Pierre-André Cornillon, Nicolas Hengartner,
Eric Matzner-Løber et Laurent Rouvière

Régression avec R

3e édition

edp sciences

ISBN (papier) : 978-2-7598-3145-6 — ISBN (ebook) : 978-2-7598-3146-3

© 2023, EDP Sciences, 17, avenue du Hoggar, BP 112, Parc d'activités de Courtaboeuf, 91944 Les Ulis Cedex A

Imprimé en France

Tous droits de traduction, d'adaptation et de reproduction par tous procédés réservés pour tous pays. Toute reproduction ou représentation intégrale ou partielle, par quelque procédé que ce soit, des pages publiées dans le présent ouvrage, faite sans l'autorisation de l'éditeur est illicite et constitue une contrefaçon. Seules sont autorisées, d'une part, les reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective, et d'autre part, les courtes citations justifiées par le caractère scientifique ou d'information de l'oeuvre dans laquelle elles sont incorporées (art. L. 122-4, L. 122-5 et L. 335-2 du Code de la propriété intellectuelle). Des photocopies payantes peuvent être réalisées avec l'accord de l'éditeur. S'adresser au : Centre français d'exploitation du droit de copie, 3, rue Hautefeuille, 75006 Paris. Tél. : 01 43 26 95 35.

Collection Pratique R

dirigée par Pierre-André Cornillon et Eric Matzner-Løber

Université Rennes-2
et ENSAE formation continue Le Cepe, France

Comité éditorial

Eva Cantoni
Institut de recherche en statistique
& Département d'économétrie
Université de Genève, Suisse

Marie Chavent
Institut de Mathématique de Bordeaux,
Centre Inria de l'université de Bordeaux
Talence, France

Rémy Drouilhet
Laboratoire Jean Kuntzmann
Université Pierre Mendes France
Grenoble, France

Ana Karina Fermin Rodriguez
Laboratoire Modal'X
Université Paris Ouest
France

François Husson
Unité Pédagogique de Mathématiques
Appliquées, Institut Agro Rennes-Angers
France

Pierre Lafaye de Micheaux
Application des Mathématiques,
Informatique, Statistique
Université Paul-Valéry Montpellier 3
France

Déjà paru dans la même collection :

Calcul parallèle avec R
Vincent Miele, Violaine Louvet, 2016
ISBN : 978-2-7598-2060-3 – EDP Sciences

Séries temporelles avec R
Yves Aragon, 2016
ISBN : 978-2-7598-1779-5 – EDP Sciences

Psychologie statistique avec R
Yvonnick Noël, 2015
ISBN : 978-2-7598-1736-8 – EDP Sciences

Réseaux bayésiens avec R
Jean-Baptiste Denis, Marco Scutati, 2014
ISBN : 978-2-7598-1198-4 – EDP Sciences

Analyse factorielle multiple avec R
Jérôme Pagès, 2013
ISBN : 978-2-7598-0963-9 – EDP Sciences

Méthodes de Monte-Carlo avec R
Christian P. Robert, George Casella, 2011
ISBN : 978-2-8178-0181-0 – Springer

REMERCIEMENTS

Cet ouvrage est l'évolution naturelle de la première édition de *Régression avec R*, elle-même issue de *Régression : Théorie et applications*.
Cette nouvelle édition s'appuie toujours sur des exemples concrets et elle n'existerait pas sans ceux-ci. Il est souvent difficile d'obtenir des données réelles pour tester ou présenter des méthodes. Et il est encore plus difficile d'obtenir l'autorisation de les publier. Or nous avons eu la chance d'avoir cette autorisation et des cohortes d'étudiants ont donc analysé des données de pollution et des données d'eucalyptus ! Nous souhaitons profiter de cette nouvelle édition pour renouveler nos sincères remerciement à M. Coron (Association Air Breizh), B. Mallet (CIRAD forêt) et J.-N. Marien (UR2PI) qui nous ont autorisé à utiliser et diffuser leurs données. Nous souhaitons bien sûr associer tous les membres de l'unité de recherche pour la productivité des plantations industrielles (UR2PI), passés ou présents. Les membres de cet organisme de recherche congolais gèrent de nombreux essais, tant génétiques que sylvicoles, et nous renvoyons toutes les personnes intéressées auprès de cet organisme ou auprès du CIRAD, département forêt (wwww.cirad.fr), qui est un des membres fondateurs et un participant actif au sein de l'UR2PI.
Plus de dix ans se sont écoulés depuis les premières versions de cet ouvrage et nous avons eu le plaisir de recevoir de nombreux retours pertinents sur les premières éditions. Les remaniements et l'ajout de nouveaux chapitres comme ceux consacrés au modèle linéaire généralisé, aux méthodes régularisées et à la régression non paramétrique nous ont incités à faire relire ces passages et à en rediscuter d'autres. Les commentaires minutieux et avisés de C. Abraham, N. Chèze, M.-L. Grisoni, P. Lafaye de Micheaux, V. Lefieux, E. Le Pennec nous ont ainsi permis d'améliorer les différents chapitres afin (nous l'espérons) de produire une nouvelle édition plus aboutie. Nous leurs adressons de chaleureux et sincères remerciements.
Nos remerciements vont également à N. Huilleret et C. Ruelle qui nous ont permis de mener à bien le projet de livre et d'édition. Enfin sans la reprise de la collection *Pratique R* par EDP Sciences, ce travail n'existerait pas. Merci donc à F. Citrini et S. Hosotte, pour leur temps, encouragements et patience. Nous remercions également EDP Sciences pour les relectures pertinentes et minutieuses de cet ouvrage.

AVANT-PROPOS

Cette nouvelle édition est une évolution de la version initiale publiée en 2009. Nous rappelons que cette première version s'inscrivait dans la continuation du livre *Régression : théorie et applications* paru chez Springer-Verlag (Paris). Cette nouvelle édition est plus qu'une mise à jour de la version initiale, la structure a été complètement repensée et de nouvelles parties sont apparues. Par ailleurs, un site web dédié au livre est proposé à l'url `https://regression-avec-r.github.io/`. On pourra notamment y trouver tous les jeux de données et les lignes de code utilisés dans chaque chapitre ainsi que les corrections des exercices.

L'objectif de cet ouvrage est de rendre accessible au plus grand nombre les différentes façons d'aborder un des problèmes auquel le statisticien est très souvent confronté : la *régression*. Les aspects théoriques et pratiques sont simultanément présentés. En effet, comme pour toute méthode statistique, il est nécessaire de comprendre précisément le modèle utilisé pour proposer des résultats pertinents sur des problèmes concrets. Si ces deux objectifs sont atteints, il sera alors aisé de transposer ces acquis à d'autres méthodes, moyennant un investissement modéré. Les grandes étapes – modélisation, estimation, choix de variables, examen de la validité du modèle choisi – restent les mêmes d'une méthode à l'autre. C'est dans cet esprit que cette nouvelle édition a été écrite.

Nous avons donc souhaité un livre avec toute la rigueur scientifique possible mais dont le contenu et les idées ne soient pas noyés dans les démonstrations et les lignes de calculs. Pour cela, seules quelques démonstrations, que nous pensons importantes, sont conservées dans le corps du texte. Les autres résultats sont démontrés à titre d'exercice. Des exercices, de difficultés variables, sont proposés en fin de chapitre. La présence de † indique des exercices plus difficiles. Des questions de cours sous la forme de QCM sont aussi proposées afin d'aider aux révisions du chapitre. Les corrections sont fournies sur le site du livre.

Afin que les connaissances acquises ne restent pas uniquement théoriques, nous avons intégré des exemples traités avec le logiciel libre R. Grâce aux commandes rapportées dans le livre, le lecteur pourra ainsi se familiariser avec le logiciel et retrouver les mêmes résultats que ceux donnés dans le livre. Nous encourageons donc les lecteurs à utiliser les données et les codes afin de s'approprier la théorie mais aussi la pratique.

Cet ouvrage s'adresse aux étudiants des filières scientifiques, élèves ingénieurs, chercheurs dans les domaines appliqués et plus généralement à toutes les personnes confrontés à un problème de régression. Il utilise notamment les notions de modèle, estimateur, biais-variance, intervalle de confiance, test... Pour les lecteurs peu à l'aise avec ces concepts, le livre de Lejeune (2004) pourra constituer une aide précieuse pour certains paragraphes. Cet ouvrage nécessite la connaissance des bases du calcul matriciel : définition d'une matrice, somme, produit, inverse, ainsi que valeurs propres et vecteurs propres. Des résultats classiques sont toutefois rappelés en annexes afin d'éviter de consulter trop souvent d'autres ouvrages.

Le livre se décompose en cinq parties, chacune constituée de deux à quatre chapitres. La première pose les fondamentaux du problème de régression et montre, à

travers quelques exemples, comment on peut l'aborder à l'aide d'un modèle linéaire simple d'abord, puis multiple. Les problèmes d'estimation ainsi que la géométrie associée à la méthode des moindres carrés sont proposés dans les deux premiers chapitres de cette partie. Le troisième chapitre propose les principaux diagnostics qui permettent de s'assurer de la validité du modèle tandis que le dernier présente quelques stratégies à envisager lorsque les hypothèses classiques du modèle linéaire ne sont pas vérifiées.
La seconde partie aborde la partie inférentielle. Il s'agit d'une des parties les plus techniques et calculatoires de l'ouvrage. Cette partie permet, entre autres, d'exposer précisément les procédures de tests et de construction d'intervalles de confiance dans le modèle linéaire. Elle décrit également les spécificités engendrées par l'utilisation de variables qualitatives dans ce modèle.
La troisième partie est consacrée à un problème désormais courant en régression : la réduction de la dimension. En effet, face à l'augmentation conséquente des données, nous sommes de plus en plus confrontés à des problèmes où le nombre de variables est (très) grand. Les techniques standards appliquées à ce type de données se révèlent souvent peu performantes et il est nécessaire de trouver des alternatives. Nous présentons tout d'abord les techniques classiques de choix de variables qui consistent à se donner un critère de performance et à rechercher à l'aide de procédures exhaustives ou pas à pas le sous-groupe de variables qui optimise le critère donné. Nous présentons ensuite les approches régularisées de type Ridge-Lasso qui consistent à trouver les estimateurs qui optimisent le critère des moindres carrés pénalisés par une fonction de la norme des paramètres. Le troisième chapitre propose de faire la régression non pas sur les variables initiales mais sur des combinaisons linéaires de celles-ci. Nous insistons sur la régression sur composantes principales (PCR) et la régression *Partial Least Square* (PLS). A ce stade, nous disposons de plusieurs algorithmes qui répondent à un même problème de régression. Il devient important de se donner une méthode qui permette d'en choisir un automatiquement (on ne laisse pas l'utilisateur décider, ce sont les données qui doivent choisir). Nous proposons un protocole basé sur la minimisation de risques empiriques calculés par des algorithmes de type validation croisée qui permet de choisir l'algorithme le plus approprié pour un problème donné.
Dans la quatrième partie, entièrement nouvelle, nous présentons le modèle linéaire généralisé. Cette partie généralise les modèles initiaux, qui permettaient de traiter uniquement le cas d'une variable à expliquer continue, à des variables à expliquer binaire (régression logistique) ou de comptage (régression de Poisson). Nous insistons uniquement sur les spécificités associées à ces types de variables, la plupart des concepts étudiés précédemment s'adaptent directement à ces cas nouveaux.
Enfin, la cinquième et dernière partie est dédiée à une introduction à l'estimation non paramétrique. Cette partie présente brièvement les estimateurs de type moyennes locales à travers les exemples des splines, estimateurs à noyau et des plus proches voisins. Elle inclut également une discussion sur les avantages et inconvénients d'une telle modélisation face aux modèles paramétriques étudiés précédemment.

Table des matières

Remerciements **vii**

Avant-Propos **ix**

I Introduction au modèle linéaire **1**

1 La régression linéaire simple **3**
1.1 Introduction . . . 3
1.1.1 Un exemple : la pollution de l'air . . . 3
1.1.2 Un second exemple : la hauteur des arbres . . . 5
1.2 Modélisation mathématique . . . 7
1.2.1 Choix du critère de qualité et distance à la droite . . . 7
1.2.2 Choix des fonctions à utiliser . . . 9
1.3 Modélisation statistique . . . 10
1.4 Estimateurs des moindres carrés . . . 11
1.4.1 Calcul des estimateurs de β_j, quelques propriétés . . . 11
1.4.2 Résidus et variance résiduelle . . . 15
1.4.3 Prévision . . . 15
1.5 Interprétations géométriques . . . 16
1.5.1 Représentation des individus . . . 16
1.5.2 Représentation des variables . . . 17
1.6 Inférence statistique . . . 19
1.7 Exemples . . . 22
1.8 Exercices . . . 29

2 La régression linéaire multiple **31**
2.1 Introduction . . . 31
2.2 Modélisation . . . 32
2.3 Estimateurs des moindres carrés . . . 34
2.3.1 Calcul de $\hat{\beta}$ . . . 35
2.3.2 Interprétation . . . 37
2.3.3 Quelques propriétés statistiques . . . 38
2.3.4 Résidus et variance résiduelle . . . 40

2.3.5 Prévision 41
2.4 Interprétation géométrique 42
2.5 Exemples 43
2.6 Exercices 47

3 Validation du modèle 51
3.1 Analyse des résidus 52
3.1.1 Les différents résidus 52
3.1.2 Ajustement individuel au modèle, valeur aberrante 53
3.1.3 Analyse de la normalité 54
3.1.4 Analyse de l'homoscédasticité 55
3.1.5 Analyse de la structure des résidus 56
3.2 Analyse de la matrice de projection 59
3.3 Autres mesures diagnostiques 60
3.4 Effet d'une variable explicative 63
3.4.1 Ajustement au modèle 63
3.4.2 Régression partielle : impact d'une variable 64
3.4.3 Résidus partiels et résidus partiels augmentés 65
3.5 Exemple : la concentration en ozone 67
3.6 Exercices 70

4 Extensions : non-inversibilité et (ou) erreurs corrélées 73
4.1 Régression ridge 73
4.1.1 Une solution historique 74
4.1.2 Minimisation des MCO pénalisés 75
4.1.3 Equivalence avec une contrainte sur la norme des coefficients 75
4.1.4 Propriétés statistiques de l'estimateur ridge $\hat{\beta}_{\text{ridge}}$ 76
4.2 Erreurs corrélées : moindres carrés généralisés 78
4.2.1 Erreurs hétéroscédastiques 79
4.2.2 Estimateur des moindres carrés généralisés 82
4.2.3 Matrice Ω inconnue 84
4.3 Exercices 85

II Inférence 89

5 Inférence dans le modèle gaussien 91
5.1 Estimateurs du maximum de vraisemblance 91
5.2 Nouvelles propriétés statistiques 92
5.3 Intervalles et régions de confiance 94
5.4 Prévision 97
5.5 Les tests d'hypothèses 98
5.5.1 Introduction 98
5.5.2 Test entre modèles emboîtés 98
5.6 Applications 102

5.7 Exercices 106
5.8 Notes 109
5.8.1 Intervalle de confiance : bootstrap 109
5.8.2 Test de Fisher pour une hypothèse linéaire quelconque . . . 112
5.8.3 Propriétés asymptotiques 114

6 Variables qualitatives : ANCOVA et ANOVA 117
6.1 Introduction 117
6.2 Analyse de la covariance 119
6.2.1 Introduction : exemple des eucalyptus 119
6.2.2 Modélisation du problème 121
6.2.3 Hypothèse gaussienne 123
6.2.4 Exemple : la concentration en ozone 124
6.2.5 Exemple : la hauteur des eucalyptus 129
6.3 Analyse de la variance à 1 facteur 131
6.3.1 Introduction 131
6.3.2 Modélisation du problème 132
6.3.3 Interprétation des contraintes 134
6.3.4 Estimation des paramètres 134
6.3.5 Hypothèse gaussienne et test d'influence du facteur 136
6.3.6 Exemple : la concentration en ozone 137
6.3.7 Une décomposition directe de la variance 142
6.4 Analyse de la variance à 2 facteurs 142
6.4.1 Introduction 142
6.4.2 Modélisation du problème 143
6.4.3 Estimation des paramètres 145
6.4.4 Analyse graphique de l'interaction 146
6.4.5 Hypothèse gaussienne et test de l'interaction 148
6.4.6 Exemple : la concentration en ozone 150
6.5 Exercices 152
6.6 Note : identifiabilité et contrastes 155

III Réduction de dimension 157

7 Choix de variables 159
7.1 Introduction 159
7.2 Choix incorrect de variables : conséquences 161
7.2.1 Biais des estimateurs 161
7.2.2 Variance des estimateurs 163
7.2.3 Erreur quadratique moyenne 163
7.2.4 Erreur quadratique moyenne de prévision 166
7.3 Critères classiques de choix de modèles 168
7.3.1 Tests entre modèles emboîtés 169
7.3.2 Le R^2 170

7.3.3 Le R^2 ajusté . . . 171
7.3.4 Le C_p de Mallows . . . 172
7.3.5 Vraisemblance et pénalisation . . . 174
7.3.6 Liens entre les critères . . . 176
7.4 Procédure de sélection . . . 178
7.4.1 Recherche exhaustive . . . 178
7.4.2 Recherche pas à pas . . . 178
7.5 Exemple : la concentration en ozone . . . 180
7.5.1 Variables explicatives quantitatives . . . 180
7.5.2 Intégration de variables qualitatives . . . 183
7.6 Exercices . . . 184
7.7 Note : C_p et biais de sélection . . . 187

8 Régularisation des moindres carrés : Ridge, Lasso et elastic-net 191
8.1 Introduction . . . 191
8.2 Problème du centrage-réduction des variables . . . 194
8.3 Ridge, lasso et elastic-net . . . 195
8.3.1 Régressions avec la package **glmnet** . . . 199
8.3.2 Interprétation géométrique . . . 202
8.3.3 Simplification quand les X sont orthogonaux . . . 203
8.3.4 Choix du paramètre de régularisation λ . . . 206
8.4 Intégration de variables qualitatives . . . 208
8.5 Exercices . . . 210
8.6 Note : lars et lasso . . . 213

9 Régression sur composantes : PCR et PLS 217
9.1 Régression sur composantes principales (PCR) . . . 218
9.1.1 Changement de base . . . 218
9.1.2 Estimateurs des MCO . . . 219
9.1.3 Choix de composantes/variables . . . 220
9.1.4 Retour aux données d'origine . . . 222
9.1.5 La régression sur composantes en pratique . . . 223
9.2 Régression aux moindres carrés partiels (PLS) . . . 225
9.2.1 Algorithmes PLS . . . 227
9.2.2 Choix de composantes/variables . . . 228
9.2.3 Retour aux données d'origine . . . 229
9.2.4 La régression PLS en pratique . . . 230
9.3 Exercices . . . 231
9.4 Notes . . . 233
9.4.1 ACP et changement de base . . . 233
9.4.2 Colinéarité parfaite : $|X'X| = 0$ . . . 235

10 Comparaison des différentes méthodes, étude de cas réels 237
10.1 Erreur de prévision et validation croisée 237
10.2 Analyse de l'ozone 241
10.2.1 Préliminaires 241
10.2.2 Méthodes et comparaison 241
10.2.3 Pour aller plus loin 245
10.2.4 Conclusion 248

IV Le modèle linéaire généralisé 249

11 Régression logistique 251
11.1 Présentation du modèle 251
11.1.1 Exemple introductif 251
11.1.2 Modélisation statistique 252
11.1.3 Variables explicatives qualitatives, interactions 255
11.2 Estimation 257
11.2.1 La vraisemblance 257
11.2.2 Calcul des estimateurs : l'algorithme IRLS 259
11.2.3 Propriétés asymptotiques de l'EMV 260
11.3 Intervalles de confiance et tests 261
11.3.1 IC et tests sur les paramètres du modèle 262
11.3.2 Test sur un sous-ensemble de paramètres 264
11.3.3 Prévision 267
11.4 Adéquation du modèle 269
11.4.1 Le modèle saturé 270
11.4.2 Tests d'adéquation de la déviance et de Pearson 272
11.4.3 Analyse des résidus 275
11.5 Choix de variables 279
11.5.1 Tests entre modèles emboîtés 279
11.5.2 Procédures automatiques 280
11.6 Exercices 282

12 Régression de Poisson 289
12.1 Le modèle linéaire généralisé (GLM) 289
12.2 Exemple : modélisation du nombre de visites 292
12.3 Régression Log-linéaire 295
12.3.1 Le modèle 295
12.3.2 Estimation 296
12.3.3 Tests et intervalles de confiance 297
12.3.4 Choix de variables 302
12.4 Exercices 303

13 Régularisation de la vraisemblance **309**
13.1 Régressions ridge, lasso et elastic-net . . . 309
13.2 Choix du paramètre de régularisation λ . . . 313
13.3 Group-lasso . . . 317
13.4 Exercices . . . 319

14 Comparaison en classification supervisée **321**
14.1 Prévision en classification supervisée . . . 321
14.2 Performance d'une règle . . . 323
14.2.1 Erreur de classification et *accuracy* . . . 326
14.2.2 Sensibilité (*recall*) et taux de faux négatifs . . . 327
14.2.3 Spécificité et taux de faux positifs . . . 327
14.2.4 Mesure sur les tables de contingence . . . 328
14.3 Performance d'un score . . . 329
14.3.1 Courbe ROC . . . 329
14.3.2 Courbe lift . . . 331
14.4 Choix du seuil . . . 332
14.4.1 Respect des proportions initiales . . . 332
14.4.2 Maximisation d'indices ad hoc . . . 332
14.4.3 Maximisation d'un coût moyen . . . 333
14.5 Analyse des données chd . . . 334
14.5.1 Les données . . . 334
14.5.2 Comparaison des algorithmes . . . 334
14.5.3 Pour aller plus loin . . . 340
14.6 Application : détection d'images publicitaires sur internet . . . 346
14.6.1 Les données . . . 346
14.6.2 Ajustement des modèles . . . 347
14.7 Exercices . . . 351

15 Données déséquilibrées **353**
15.1 Données déséquilibrées et modèle logistique . . . 353
15.1.1 Un exemple . . . 353
15.1.2 Rééquilibrage pour le modèle logistique . . . 355
15.1.3 Exemples de schéma de rééquilibrage . . . 356
15.2 Stratégies pour données déséquilibrées . . . 361
15.2.1 Quelques méthodes de rééquilibrage . . . 361
15.2.2 Critères pour données déséquilibrées . . . 366
15.3 Choisir un algorithme de rééquilibrage . . . 370
15.3.1 Rééquilibrage et validation croisée . . . 370
15.3.2 Application aux données d'images publicitaires . . . 372
15.4 Exercices . . . 376

V Introduction à la régression non paramétrique 379

16 Introduction à la régression spline 381
16.1 Introduction 381
16.2 Régression spline 385
16.2.1 Introduction 385
16.2.2 Spline de régression 386
16.3 Spline de lissage 390
16.4 Exercices 393

17 Estimateurs à noyau et k plus proches voisins 395
17.1 Introduction 395
17.2 Estimateurs par moyennes locales 398
17.2.1 Estimateurs à noyau 398
17.2.2 Les k plus proches voisins 402
17.3 Choix des paramètres de lissage 403
17.4 Ecriture multivariée et fléau de la dimension 406
17.4.1 Ecriture multivariée 406
17.4.2 Biais et variance 407
17.4.3 Fléau de la dimension 409
17.5 Exercices 411

A Rappels 415
A.1 Rappels d'algèbre 415
A.2 Rappels de probabilités 418

Bibliographie 419

Index 423

Notations 431

Fonctions et packages R 433

Première partie

Introduction au modèle linéaire

Chapitre 1

La régression linéaire simple

1.1 Introduction

L'origine du mot régression vient de Sir Francis Galton. En 1885, travaillant sur l'hérédité, il chercha à expliquer la taille des fils en fonction de celle des pères. Il constata que lorsque le père était plus grand que la moyenne, *taller than mediocrity*, son fils avait tendance à être plus petit que lui et, *a contrario*, que lorsque le père était plus petit que la moyenne, *shorter than mediocrity*, son fils avait tendance à être plus grand que lui. Ces résultats l'ont conduit à considérer sa théorie de *regression toward mediocrity*. Cependant, l'analyse de causalité entre plusieurs variables est plus ancienne et remonte au milieu du XVIIIe siècle. En 1757, R. Boscovich, né à Ragusa, l'actuelle Dubrovnik, proposa une méthode minimisant la somme des valeurs absolues entre un modèle de causalité et les observations. Ensuite Legendre, dans son célèbre article de 1805, « Nouvelles méthodes pour la détermination des orbites des comètes », introduisit la méthode d'estimation par moindres carrés des coefficients d'un modèle de causalité et donna le nom à la méthode. Parallèlement, Gauss publia en 1809 un travail sur le mouvement des corps célestes qui contenait un développement de la méthode des moindres carrés, qu'il affirmait utiliser depuis 1795 (Birkes & Dodge, 1993).
Dans ce chapitre, nous allons analyser la régression linéaire simple : nous pouvons la voir comme une technique statistique permettant de modéliser la relation linéaire entre une variable explicative (notée X) et une variable à expliquer (notée Y). Cette présentation dans un cas simple va nous permettre de bien comprendre les enjeux de cette méthode, les problèmes posés et les réponses apportées.

1.1.1 Un exemple : la pollution de l'air

La pollution de l'air constitue actuellement une des préoccupations majeures de santé publique. De nombreuses études épidémiologiques ont permis de mettre en évidence l'influence sur la santé de certains composés chimiques comme le dioxyde de souffre (SO_2), le dioxyde d'azote (NO_2), l'ozone (O_3) ou des particules sous

forme de poussières contenues dans l'air. L'influence de cette pollution est notable sur les personnes sensibles (nouveau-nés, asthmatiques, personnes âgées). La prévision des pics de concentration de ces composés est donc importante. Nous nous intéressons plus particulièrement à la concentration en ozone. Nous possédons quelques connaissances *a priori* sur la manière dont se forme l'ozone, grâce aux lois régissant les équilibres chimiques. Un des catalyseurs de l'ozone est le rayonnement, cette variable est plus difficile à prévoir que la température et nous allons étudier la concentration de l'ozone, qui est fonction de la température ; plus la température est élevée, plus la concentration en ozone est importante. Cette relation très vague doit être améliorée afin de pouvoir prédire les pics d'ozone.
Afin de mieux comprendre ce phénomène, l'association Air Breizh (surveillance de la qualité de l'air en Bretagne) mesure depuis 1994 la concentration en O_3 (en μg/ml) toutes les 10 minutes. Le maximum journalier de la concentration en O_3 sera noté `O3`. Air Breizh collecte également à certaines heures de la journée des données météorologiques comme la température, la nébulosité, le vent... Les données sont disponibles en ligne (voir Avant-propos). Le tableau suivant donne les 5 premières mesures effectuées.

Individu	`O3`	`T12`
1	63.6	13.4
2	89.6	15
3	79	7.9
4	81.2	13.1
5	88	14.1

Tableau 1.1 – 5 données de température à 12 h et teneur maximale en ozone.

Nous allons donc chercher à expliquer le maximum de `O3` de la journée par la température à 12 h. Le but de cette régression est double :
— ajuster un modèle pour expliquer la concentration en `O3` en fonction de `T12` ;
— prédire les valeurs de concentration en `O3` pour de nouvelles valeurs de `T12`.
Avant toute analyse, il est intéressant de représenter les données.

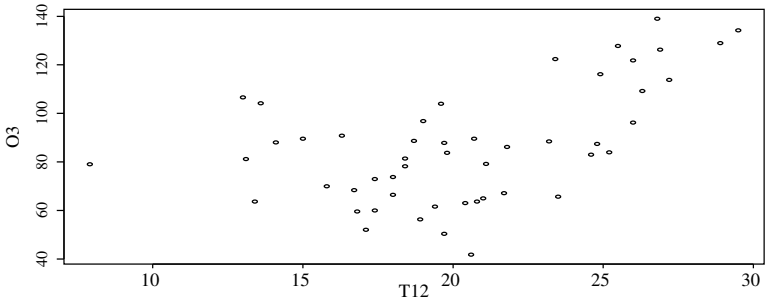


Fig. 1.1 – 50 données journalières de température et `O3`.

Chaque point du graphique (fig.1.1) représente, pour un jour donné, une mesure de la température à 12 h et le pic d'ozone de la journée.
Pour analyser la relation entre les x_i (température) et les y_i (ozone), nous allons chercher une fonction f telle que

$$y_i \approx f(x_i).$$

Pour définir $\approx$, il faut donner un critère quantifiant la qualité de l'ajustement de la fonction f aux données et une classe de fonctions $\mathcal{G}$ dans laquelle est supposée se trouver la vraie fonction inconnue. Le problème mathématique peut s'écrire de la façon suivante :

$$\underset{f \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} l(y_i - f(x_i)), \tag{1.1}$$

où n représente le nombre de données à analyser et $l(.)$ est appelée *fonction de coût* ou encore *fonction de perte*.

1.1.2 Un second exemple : la hauteur des arbres

Cet exemple utilise des données fournies par l'UR2PI et le CIRAD forêt (voir Remerciements). Lorsque le forestier évalue la vigueur d'une forêt, il considère souvent la hauteur des arbres qui la composent. Plus les arbres sont hauts, plus la forêt ou la plantation produit. Afin de calculer le volume de l'arbre, il est nécessaire d'avoir sa hauteur et d'utiliser ensuite une formule du type « tronc de cône ». Cependant, mesurer la hauteur d'un arbre d'une vingtaine de mètres n'est pas aisé et demande un dendromètre. Ce type d'appareil mesure un angle entre le sol et le sommet de l'arbre. Il nécessite donc une vision claire de la cime de l'arbre et un recul assez grand afin d'avoir une mesure précise de l'angle et donc de la hauteur.
Dans certains cas, il est impossible de mesurer la hauteur, car ces deux conditions ne sont pas réunies, ou la mesure demande quelquefois trop de temps ou encore le forestier n'a pas de dendromètre. Il est alors nécessaire d'estimer la hauteur grâce à une mesure simple, la mesure de la circonférence à 1 mètre 30 du sol.
Nous possédons des mesures sur des eucalyptus dans une parcelle plantée et nous souhaitons à partir de ces mesures élaborer un modèle de prévision de la hauteur. Les eucalyptus étant plantés pour servir de matière première dans la pâte à papier, ils sont vendus au volume de bois. Il est donc important de connaître le volume et la hauteur, afin d'évaluer la réserve en matière première dans la plantation (ou volume sur pied total). Les surfaces plantées sont énormes, il n'est pas question de prendre trop de temps pour la mesure et prévoir la hauteur par la circonférence est une méthode permettant la prévision du volume sur pied. La parcelle d'intérêt est constituée d'eucalyptus de 6 ans, âge de « maturité » des eucalyptus, c'est-à-dire l'âge en fin de rotation avant la coupe. Dans cette parcelle, nous avons alors mesuré $n = 1429$ couples circonférence-hauteur. Le tableau suivant donne les 5 premières mesures effectuées.

Individu	`ht`	`circ`
1	18.25	36
2	19.75	42
3	16.50	33
4	18.25	39
5	19.50	43

Tableau 1.2 – Hauteur et circonférence (`ht` et `circ`) des 5 premiers eucalyptus.

Nous souhaitons donc expliquer la hauteur par la circonférence. Avant toute modélisation, nous représentons les données. Chaque point du graphique 1.2 représente une mesure du couple circonférence/hauteur sur un eucalyptus.

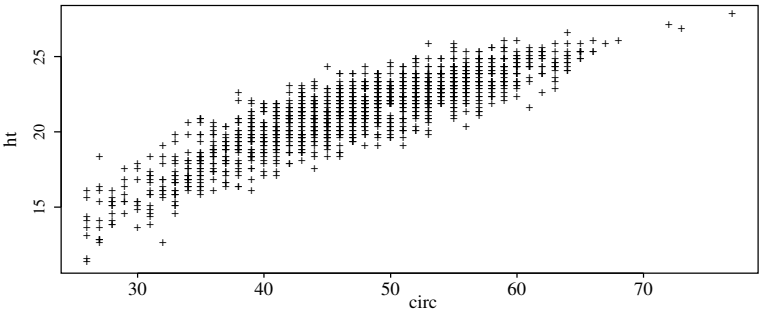


Fig. 1.2 – Représentation des mesures pour les $n = 1429$ eucalyptus mesurés.

Pour prévoir la hauteur en fonction de la circonférence, nous allons donc chercher une fonction f telle que

$$y_i \approx f(x_i)$$

pour chaque mesure $i \in \{1, \ldots, 1429\}$.
A nouveau, afin de quantifier le symbole $\approx$, nous allons choisir une classe de fonctions $\mathcal{G}$. Cette classe représente toutes les fonctions d'ajustement possible pour modéliser la hauteur en fonction de la circonférence. Puis nous cherchons la fonction de $\mathcal{G}$ qui soit la plus proche possible des données selon une fonction de coût. Cela s'écrit

$$\underset{f \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} l(y_i - f(x_i)),$$

où n représente le nombre de données à analyser et $l(.)$ la *fonction de coût*.

Remarque
Le calcul du volume proposé ici est donc fait en deux étapes : dans la première on estime la hauteur et dans la seconde on utilise une formule de type « tronc de cône » pour calculer le volume avec la hauteur estimée et la circonférence. Une autre méthode de calcul de volume consiste à ne pas utiliser de formule incluant

la hauteur et prévoir directement le volume en une seule étape. Pour cela il faut calibrer le volume en fonction de la circonférence et il faut donc la mesure de nombreux volumes en fonction de circonférences, ce qui est très coûteux et difficile à réactualiser.

1.2 Modélisation mathématique

Nous venons de voir que le problème mathématique peut s'écrire de la façon suivante (voir équation 1.1) :

$$\underset{f \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} l(y_i - f(x_i)),$$

où $l(.)$ est appelée *fonction de coût* et $\mathcal{G}$ un ensemble de fonctions données. Dans la suite de cette section, nous allons discuter du choix de la fonction de coût et de l'ensemble $\mathcal{G}$. Nous présenterons des graphiques illustratifs bâtis à partir de 10 données fictives de température et de concentration en ozone.

1.2.1 Choix du critère de qualité et distance à la droite

De nombreuses fonctions de coût $l(.)$ existent, mais les deux principales utilisées sont les suivantes :

— $l(u) = u^2$ coût quadratique ;
— $l(u) = |u|$ coût absolu.

Ces deux fonctions sont représentées sur le graphique 1.3 :

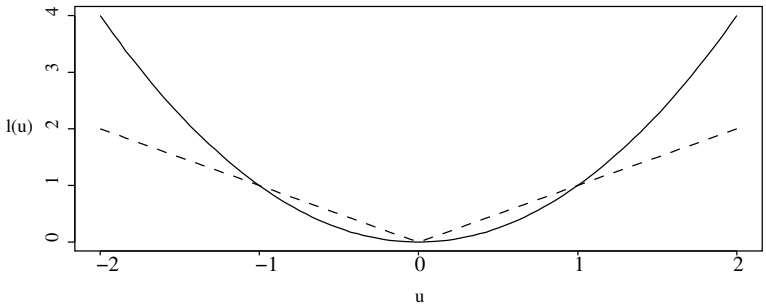


Fig. 1.3 – Coût absolu (pointillés) et coût quadratique (trait plein).

Ces fonctions sont positives, symétriques, elles donnent donc la même valeur lorsque l'erreur est positive ou négative et s'annulent lorsque u vaut zéro. De manière plus générale, l n'est pas obligatoirement symétrique mais est une fonction convexe positive et peut être vue comme la distance entre une observation (x_i, y_i) et son point correspondant sur la droite $(x_i, f(x_i))$ (voir fig. 1.4).

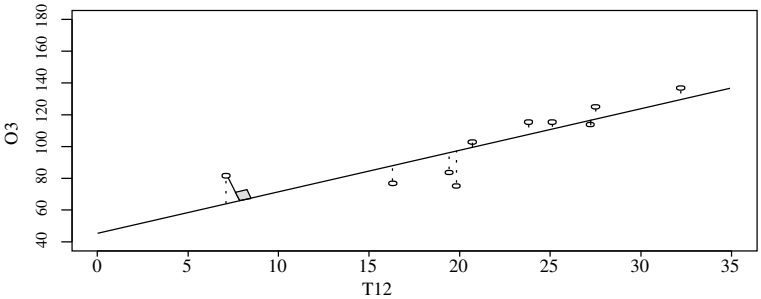


Fig. 1.4 – Distances à la droite : coût absolu (pointillés) et distance d'un point à une droite.

Par point correspondant, nous entendons « évalué » à la même valeur x_i. Nous aurions pu prendre comme critère à minimiser la somme des distances des points (x_i, y_i) à la droite[1] (voir fig. 1.4), mais ce type de distance n'entre pas dans le cadre des fonctions de coût puisqu'au point (x_i, y_i) correspond sur la droite un point $(x'_i, f(x'_i))$ d'abscisse et d'ordonnée différentes.
Il est évident que, par rapport au coût absolu, le coût quadratique accorde une importance plus grande aux points qui restent éloignés de la droite ajustée, la distance étant élevée au carré (voir fig. 1.3). Sur l'exemple fictif, dans la classe $\mathcal{G}$ des fonctions linéaires, nous allons minimiser $\sum_{i=1}^{n}(y_i - f(x_i))^2$ (coût quadratique) et $\sum_{i=1}^{n}|y_i - f(x_i)|$ (coût absolu). Les droites ajustées sont représentées sur le graphique ci-dessous :

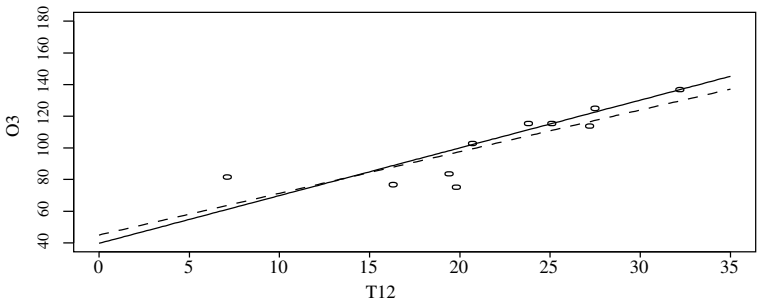


Fig. 1.5 – 10 données fictives de température et O3, régressions avec un coût absolu (trait plein) et quadratique (pointillés).

La droite ajustée avec un coût quadratique propose un compromis où aucun point n'est très éloigné de la droite : le coût quadratique est sensible aux points aberrants qui sont éloignés de la droite. Ainsi (fig. 1.5) le premier point d'abscisse approximative 7°C est assez éloigné des autres. La droite ajustée avec un coût quadratique lui accorde une plus grosse importance que l'autre droite et passe relativement donc plus près de lui. En enlevant ce point (de manière imaginaire),

1. La distance d'un point à une droite est la longueur de la perpendiculaire à cette droite passant par ce point.

la droite ajustée risque d'être très différente : le point est dit influent et le coût quadratique peu robuste. Le coût absolu est plus robuste et la modification d'une observation modifie moins la droite ajustée. Les notions de points influents, points aberrants, seront approfondies au chapitre 3.
Malgré cette non-robustesse, le coût quadratique est le coût le plus souvent utilisé, pour plusieurs raisons : historique, calculabilité, propriétés mathématiques. En 1800, il n'existait pas d'ordinateur et l'utilisation du coût quadratique permettait de calculer explicitement les estimateurs à partir des données. A propos de l'utilisation d'autres fonctions de coût, voici ce que disait Gauss (1809) : « Mais de tous ces principes, celui des moindres carrés est le plus simple : avec les autres, nous serions conduits aux calculs les plus complexes ». En conclusion, *seul le coût quadratique sera automatiquement utilisé dans la suite de ce livre, sauf mention contraire*. Les lecteurs intéressés par le coût absolu peuvent consulter le livre de Dodge & Rousson (2004).

1.2.2 Choix des fonctions à utiliser

Si la classe $\mathcal{G}$ est trop large, par exemple la classe des fonctions continues ($\mathcal{C}_0$), un grand nombre de fonctions de cette classe minimisent le critère (1.1). Ainsi toutes les fonctions de la classe qui passent par tous les points (interpolation), quand c'est possible, annulent la quantité $\sum_{i=1}^{n} l(y_i - f(x_i))$.

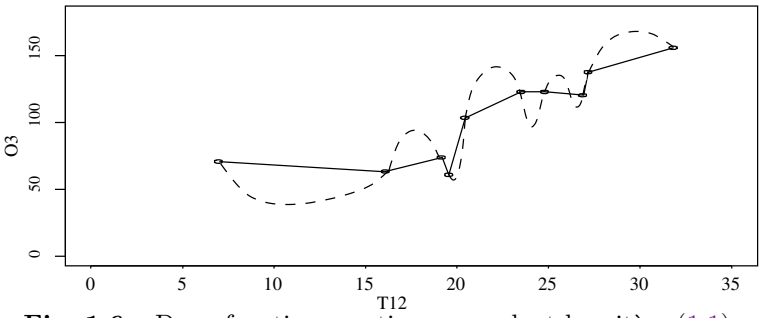


Fig. 1.6 – Deux fonctions continues annulant le critère (1.1).

La fonction continue tracée en pointillés sur la figure (fig. 1.6) semble inappropriée bien qu'elle annule le critère (1.1). La fonction continue tracée en traits pleins annule aussi le critère (1.1). D'autres fonctions continues annulent ce critère, donc la classe des fonctions continues est trop vaste. Ces fonctions passent par tous les points et c'est là leur principal défaut. Nous souhaitons plutôt une courbe, ne passant pas par tous les points, mais possédant un trajet harmonieux, sans trop de détours. Bien sûr le trajet sans aucun détour est la ligne droite et la classe $\mathcal{G}$ la plus simple sera l'ensemble des fonctions affines. Par abus de langage, on emploie le terme de fonctions linéaires. D'autres classes de fonctions peuvent être choisies et ce choix est en général dicté par une connaissance *a priori* du phénomène et (ou) par l'observation des données.

Ainsi une étude de régression linéaire simple débute toujours par une représentation graphique des observations (x, y), appelée nuage de points. Cette première représentation permet de savoir si le modèle linéaire est pertinent. Le graphique (fig. 1.7) représente trois nuages de points différents.

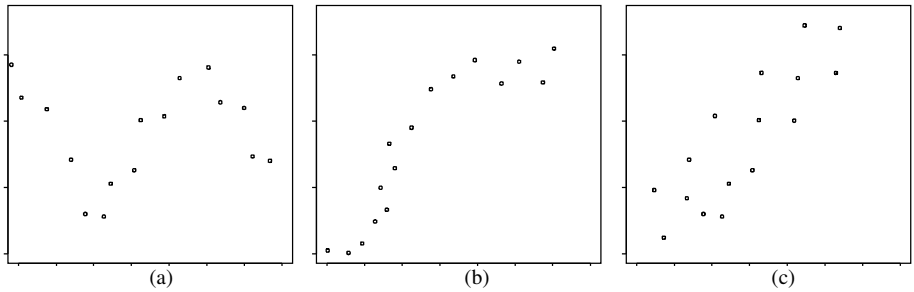


Fig. 1.7 – Exemples de nuages de points.

Au vu du graphique, il semble inadéquat de proposer une régression linéaire pour les graphiques (a) et (b), le tracé présentant une forme sinusoïdale ou sigmoïdale. Par contre, la modélisation par une droite de la relation entre X et Y pour le graphique (c) semble correspondre à la réalité de la liaison. Dans la suite de ce chapitre, nous prendrons $\mathcal{G} = \left\{ f : f(x) = ax + b, \quad (a, b) \in \mathbb{R}^2 \right\}$.

1.3 Modélisation statistique

Lorsque nous ajustons par une droite les données, nous supposons implicitement que leur relation était de la forme

$$Y = \beta_1 + \beta_2 X.$$

Dans l'exemple de l'ozone, nous supposons donc un modèle où la concentration d'ozone dépend linéairement de la température. Nous savons pertinemment que toutes les observations mesurées ne sont pas sur la droite. D'une part, il est irréaliste de croire que la concentration de l'ozone dépend linéairement de la température et de la température seulement. D'autre part, les mesures effectuées dépendent de la précision de l'appareil de mesure, de l'opérateur et il peut arriver que pour des valeurs identiques de la variable X, nous observions des valeurs différentes pour Y.
Nous supposons alors que la concentration d'ozone dépend linéairement de la température mais cette liaison est perturbée par un « bruit ». Nous supposons en fait que les données suivent le modèle suivant :

$$Y = \beta_1 + \beta_2 X + \varepsilon. \tag{1.2}$$

L'équation (1.2) est appelée **modèle de régression linéaire** et dans ce cas précis **modèle de régression linéaire simple**. Les β_j, appelés les paramètres du modèle (constante de régression et coefficient de régression), sont fixes mais inconnus,

et nous voulons les estimer. La quantité notée ε est appelée bruit, ou erreur, et est aléatoire et inconnue.
Afin d'estimer les paramètres inconnus du modèle, nous mesurons dans le cadre de la régression simple une seule variable explicative ou variable exogène X et une variable à expliquer ou variable endogène Y. La variable X est souvent considérée comme non aléatoire au contraire de Y. Nous mesurons alors n observations de la variable X, notées x_i, où i varie de 1 à n, et n valeurs de la variable à expliquer Y notées y_i.
Nous supposons que nous avons collecté n couples de données (x_i, y_i) où y_i est la réalisation de la variable aléatoire Y_i. Par abus de notation, nous confondrons la variable aléatoire Y_i et sa réalisation, l'observation y_i. Avec la notation ε_i, nous confondrons la variable aléatoire avec sa réalisation. Suivant le modèle (1.2), nous pouvons écrire

$$y_i = \beta_1 + \beta_2 x_i + \varepsilon_i, \qquad i = 1, \cdots, n$$

où
- les x_i sont des observations connues non aléatoires et non toutes identiques ;
- les paramètres β_j, $j = 1, 2$ du modèle sont inconnus ;
- les ε_i sont les réalisations inconnues d'une variable aléatoire ;
- les y_i sont les observations d'une variable aléatoire.

1.4 Estimateurs des moindres carrés

Définition 1.1 (estimateurs des MC)
On appelle estimateurs des moindres carrés (MC) de β_1 et β_2, les estimateurs $\hat{\beta}_1$ et $\hat{\beta}_2$ obtenus par minimisation de la quantité

$$S(\beta_1, \beta_2) = \sum_{i=1}^{n} (y_i - \beta_1 - \beta_2 x_i)^2 = \|Y - \beta_1 \mathbb{1} - \beta_2 X\|^2,$$

où $\mathbb{1}$ est le vecteur de $\mathbb{R}^n$ dont tous les coefficients valent 1. Les estimateurs peuvent également s'écrire sous la forme suivante :

$$(\hat{\beta}_1, \hat{\beta}_2) = \underset{(\beta_1, \beta_2) \in \mathbb{R} \times \mathbb{R}}{\operatorname{argmin}} S(\beta_1, \beta_2).$$

1.4.1 Calcul des estimateurs de β_j, quelques propriétés

La fonction $S(\beta_1, \beta_2)$ est strictement convexe. Si elle admet un point singulier, celui-ci correspond à l'unique minimum. Annulons les dérivées partielles, nous obtenons un système d'équations appelées équations normales :

$$\begin{cases} \dfrac{\partial S(\hat{\beta}_1, \hat{\beta}_2)}{\partial \beta_1} & = & -2 \displaystyle\sum_{i=1}^{n} (y_i - \hat{\beta}_1 - \hat{\beta}_2 x_i) = 0, \\ \dfrac{\partial S(\hat{\beta}_1, \hat{\beta}_2)}{\partial \beta_2} & = & -2 \displaystyle\sum_{i=1}^{n} x_i (y_i - \hat{\beta}_1 - \hat{\beta}_2 x_i) = 0. \end{cases}$$

La première équation donne

$$\hat{\beta}_1 n + \hat{\beta}_2 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

et nous avons un estimateur de l'ordonnée à l'origine

$$\hat{\beta}_1 = \bar{y} - \hat{\beta}_2 \bar{x}, \tag{1.3}$$

où $\bar{x} = \sum_{i=1}^{n} x_i/n$. La seconde équation donne

$$\hat{\beta}_1 \sum_{i=1}^{n} x_i + \hat{\beta}_2 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i.$$

En remplaçant $\hat{\beta}_1$ par son expression (1.3), nous avons une première écriture de

$$\hat{\beta}_2 = \frac{\sum x_i y_i - \sum x_i \bar{y}}{\sum x_i^2 - \sum x_i \bar{x}},$$

et en utilisant astucieusement la nullité de la somme $\sum(x_i - \bar{x})$, nous avons d'autres écritures pour l'estimateur de la pente de la droite

$$\hat{\beta}_2 = \frac{\sum x_i(y_i - \bar{y})}{\sum x_i(x_i - \bar{x})} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})(x_i - \bar{x})} = \frac{\sum(x_i - \bar{x})y_i}{\sum(x_i - \bar{x})^2}. \tag{1.4}$$

Pour obtenir ce résultat, nous supposons qu'il existe au moins deux points d'abscisses différentes. Cette hypothèse notée $\mathcal{H}_1$ s'écrit $x_i \neq x_j$ pour au moins deux individus. Elle permet d'obtenir l'unicité des coefficients estimés $\hat{\beta}_1, \hat{\beta}_2$.

Une fois déterminés les estimateurs $\hat{\beta}_1$ et $\hat{\beta}_2$, nous pouvons estimer la droite de régression par la formule

$$\hat{Y} = \hat{\beta}_1 + \hat{\beta}_2 X.$$

Si nous évaluons la droite aux points x_i ayant servi à estimer les paramètres, nous obtenons des $\hat{y}_i$ et ces valeurs sont appelées les valeurs ajustées. Si nous évaluons la droite en d'autres points, les valeurs obtenues seront appelées les valeurs prévues ou prévisions. Représentons les points initiaux et la droite de régression estimée. La droite de régression passe par le centre de gravité du nuage de points $(\bar{x}, \bar{y})$ (fig. 1.8) comme l'indique l 'équation (1.3).

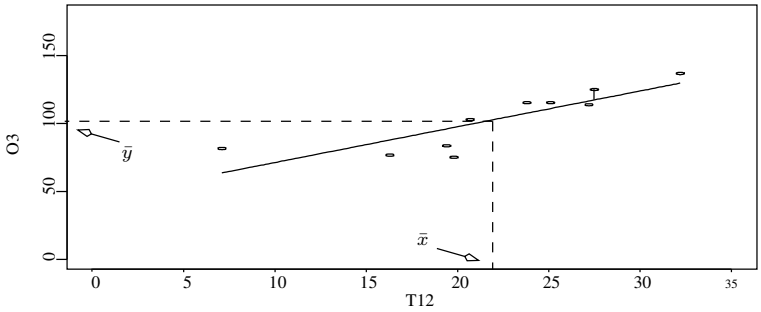


Fig. 1.8 – Nuage de points, droite de régression et centre de gravité.

Nous avons réalisé une expérience et avons mesuré n valeurs (x_i, y_i). A partir de ces n valeurs, nous avons obtenu un estimateur de β_1 et de β_2. Si nous refaisions une expérience, nous mesurerions n nouveaux couples de données (x_i, y_i). A partir de ces données, nous aurions un nouvel estimateur de β_1 et de β_2. Les estimateurs sont fonction des données mesurées et changent donc avec les observations collectées (fig. 1.9). Les vraies valeurs de β_1 et β_2 sont inconnues et ne changent pas.

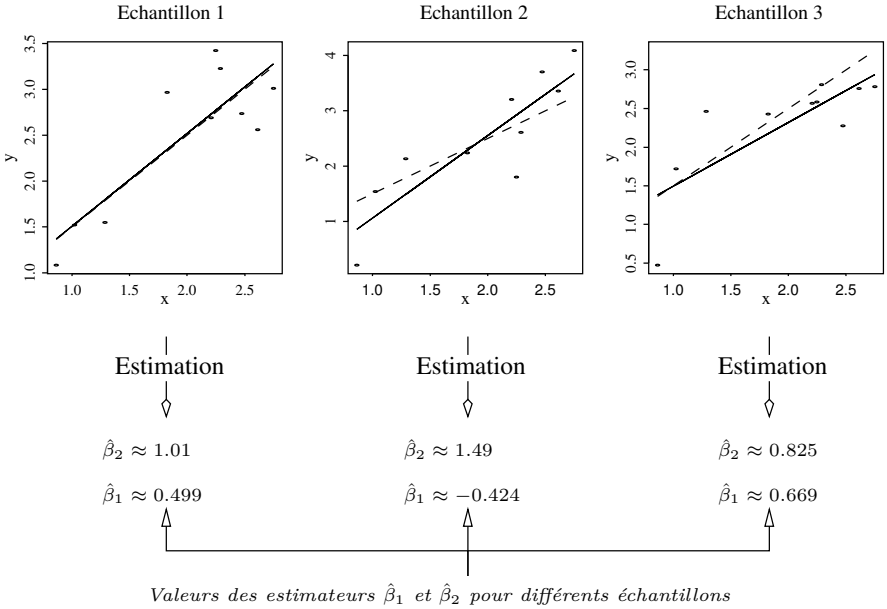


Fig. 1.9 – Exemple de la variabilité des estimations. Le vrai modèle est $Y = X + 0.5 + \varepsilon$, où ε est choisi comme suivant une loi $\mathcal{N}(0, 0.25)$. Nous avons ici 3 répétitions de la mesure de 10 points (x_i, y_i), ou 3 échantillons de taille 10. Le trait en pointillé représente la vraie droite de régression et le trait plein son estimation.

Le statisticien cherche en général à vérifier que les estimateurs utilisés admettent certaines propriétés comme :

— un estimateur $\hat{\beta}$ est-il sans biais ? Par définition $\hat{\beta}$ est sans biais si $\mathbb{E}(\hat{\beta}) = \beta$. En moyenne sur toutes les expériences possibles de taille n, l'estimateur $\hat{\beta}$ moyen sera égal à la valeur inconnue du paramètre. En français, cela signifie qu'en moyenne $\hat{\beta}$ « tombe » sur β ;
— un estimateur $\hat{\beta}$ est-il de variance minimale parmi les estimateurs d'une classe définie ? En d'autres termes, parmi tous les estimateurs de la classe, l'estimateur utilisé admet-il parmi toutes les expériences la plus petite variabilité ?

Pour cela, nous supposons une seconde hypothèse notée $\mathcal{H}_2$: les erreurs sont centrées, de même variance (homoscédasticité) et non corrélées entre elles. Elle permet de calculer les propriétés statistiques des estimateurs. $\mathcal{H}_2$: $\mathbb{E}(\varepsilon_i) = 0$, pour $i = 1, \cdots, n$ et $\mathrm{Cov}(\varepsilon_i, \varepsilon_j) = \delta_{ij}\sigma^2$, où $\mathbb{E}(\varepsilon)$ est l'espérance de ε, $\mathrm{Cov}(\varepsilon_i, \varepsilon_j)$ est la covariance entre ε_i et ε_j et $\delta_{ij} = 1$ lorsque $i = j$ et $\delta_{ij} = 0$ lorsque $i \neq j$. Nous avons la première propriété de ces estimateurs (voir exercice 1.2).

Proposition 1.1 (Biais des estimateurs)
$\hat{\beta}_1$ et $\hat{\beta}_2$ estiment sans biais β_1 et β_2, c'est-à-dire que $\mathbb{E}(\hat{\beta}_1) = \beta_1$ et $\mathbb{E}(\hat{\beta}_2) = \beta_2$.

Les estimateurs $\hat{\beta}_1$ et $\hat{\beta}_2$ sont sans biais, nous allons nous intéresser à leur variance. Afin de montrer que ces estimateurs sont de variances minimales dans leur classe, nous allons d'abord calculer leur variance (voir exercices 1.3, 1.4). C'est l'objet de la prochaine proposition.

Proposition 1.2 (Variances de $\hat{\beta}_1$ et $\hat{\beta}_2$)
Les variances et covariance des estimateurs des paramètres valent :

$$\begin{aligned}
\mathrm{V}(\hat{\beta}_2) &= \frac{\sigma^2}{\sum(x_i - \bar{x})^2} \\
\mathrm{V}(\hat{\beta}_1) &= \frac{\sigma^2 \sum x_i^2}{n \sum(x_i - \bar{x})^2} \\
\mathrm{Cov}(\hat{\beta}_1, \hat{\beta}_2) &= -\frac{\sigma^2 \bar{x}}{\sum(x_i - \bar{x})^2}.
\end{aligned}$$

Cette proposition nous permet d'envisager la précision de l'estimation en utilisant la variance. Plus la variance est faible, plus l'estimateur sera précis. Pour avoir des variances petites, il faut avoir un numérateur petit et (ou) un dénominateur grand. Les estimateurs seront donc de faibles variances lorsque :
— la variance σ^2 est faible. Cela signifie que la variance de Y est faible et donc les mesures sont proches de la droite à estimer ;
— la quantité $\sum(x_i - \bar{x})^2$ est grande, les mesures x_i doivent être dispersées autour de leur moyenne ;
— la quantité $\sum x_i^2$ ne doit pas être trop grande, les points doivent avoir une faible moyenne en valeur absolue. En effet, nous avons

$$\frac{\sum x_i^2}{\sum(x_i - \bar{x})^2} = \frac{\sum x_i^2 - n\bar{x}^2 + n\bar{x}^2}{\sum(x_i - \bar{x})^2} = 1 + \frac{n\bar{x}^2}{\sum(x_i - \bar{x})^2}.$$

L'équation (1.3) indique que la droite des MC passe par le centre de gravité du nuage $(\bar{x}, \bar{y})$. Supposons $\bar{x}$ positif, alors si nous augmentons la pente, l'ordonnée à l'origine va diminuer et vice versa. Nous retrouvons donc le signe négatif pour la covariance entre $\hat{\beta}_1$ et $\hat{\beta}_2$.
Nous terminons cette partie concernant les propriétés par le théorème de Gauss-Markov qui indique que, parmi tous les estimateurs linéaires sans biais, les estimateurs des MC possèdent la plus petite variance (voir exercice 1.5).

Théorème 1.1 (Gauss-Markov)
Parmi les estimateurs sans biais linéaires en Y, les estimateurs $\hat{\beta}_j$ sont de variance minimale.

1.4.2 Résidus et variance résiduelle

Nous avons estimé β_1 et β_2. La variance σ^2 des ε_i est le dernier paramètre inconnu à estimer. Pour cela, nous allons utiliser les résidus : ce sont des estimateurs des erreurs inconnues ε_i.

Définition 1.2 (Résidus)
Les résidus sont définis par

$$\hat{\varepsilon}_i = y_i - \hat{y}_i$$

où $\hat{y}_i$ est la valeur ajustée de y_i par le modèle, c'est-à-dire $\hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 x_i$.

Nous avons la propriété suivante (voir exercice 1.6).

Proposition 1.3
Dans un modèle de régression linéaire simple, la somme des résidus est nulle.

Intéressons-nous maintenant à l'estimation de σ^2 et construisons un estimateur sans biais $\hat{\sigma}^2$ (voir exercice 1.7) :

Proposition 1.4 (Estimateur de la variance du bruit)
La statistique $\hat{\sigma}^2 = \sum_{i=1}^{n} \hat{\varepsilon}_i^2/(n-2)$ est un estimateur sans biais de σ^2.

1.4.3 Prévision

Un des buts de la régression est de proposer des prévisions pour la variable à expliquer Y. Soit x_{n+1} une nouvelle valeur de la variable X, nous voulons prédire y_{n+1}. Le modèle indique que

$$y_{n+1} = \beta_1 + \beta_2 x_{n+1} + \varepsilon_{n+1}$$

avec $\mathbb{E}(\varepsilon_{n+1}) = 0$, $\mathrm{V}(\varepsilon_{n+1}) = \sigma^2$ et $\mathrm{Cov}(\varepsilon_{n+1}, \varepsilon_i) = 0$ pour $i = 1, \cdots, n$. Nous pouvons prédire la valeur correspondante grâce au modèle estimé

$$\hat{y}_{n+1}^p = \hat{\beta}_1 + \hat{\beta}_2 x_{n+1}.$$

En utilisant la notation $\hat{y}^p_{n+1}$ nous souhaitons insister sur la notion de prévision : la valeur pour laquelle nous effectuons la prévision, ici la $(n+1)^{\text{e}}$, n'a pas servi dans le calcul des estimateurs. Remarquons que cette quantité sera différente de la valeur ajustée, notée $\hat{y}_i$, qui elle fait intervenir la i^{e} observation.

Deux types d'erreurs vont entacher notre prévision, l'une due à la non-connaissance de ε_{n+1} et l'autre due à l'estimation des paramètres.

Proposition 1.5 (Variance de la prévision $\hat{y}^p_{n+1}$)
La variance de la valeur prévue de $\hat{y}^p_{n+1}$ vaut

$$\mathrm{V}\left(\hat{y}^p_{n+1}\right) = \sigma^2 \left(\frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum (x_i - \bar{x})^2}\right).$$

La variance de $\hat{y}^p_{n+1}$ (voir exercice 1.8) nous donne une idée de la stabilité de l'estimation. En prévision, on s'intéresse généralement à l'erreur que l'on commet entre la vraie valeur à prévoir y_{n+1} et celle que l'on prévoit $\hat{y}^p_{n+1}$. L'erreur peut être simplement résumée par la différence entre ces deux valeurs, c'est ce que nous appellerons l'erreur de prévision. Cette erreur de prévision permet de quantifier la capacité du modèle à prévoir. Nous avons sur ce thème la proposition suivante (voir exercice 1.8).

Proposition 1.6 (Erreur de prévision)
L'erreur de prévision, définie par $\hat{\varepsilon}^p_{n+1} = y_{n+1} - \hat{y}^p_{n+1}$ satisfait les propriétés suivantes :

$$\begin{aligned}
\mathbb{E}(\hat{\varepsilon}^p_{n+1}) &= 0 \\
\mathrm{V}(\hat{\varepsilon}^p_{n+1}) &= \sigma^2 \left(1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum (x_i - \bar{x})^2}\right).
\end{aligned}$$

Remarque
La variance augmente lorsque x_{n+1} s'éloigne du centre de gravité du nuage. Effectuer une prévision lorsque x_{n+1} est « loin » de $\bar{x}$ est donc périlleux, la variance de l'erreur de prévision peut alors être très grande !

1.5 Interprétations géométriques

1.5.1 Représentation des individus

Pour chaque individu, ou observation, nous mesurons une valeur x_i et une valeur y_i. Une observation peut donc être représentée dans le plan, nous dirons alors que $\mathbb{R}^2$ est l'espace des observations. $\hat{\beta}_1$ correspond à l'ordonnée à l'origine alors que $\hat{\beta}_2$ représente la pente de la droite ajustée. Cette droite minimise la somme des carrés des distances verticales des points du nuage à la droite ajustée (fig. 1.10).

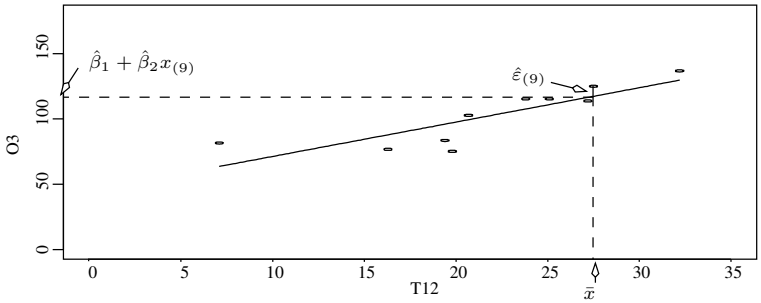


Fig. 1.10 – Représentation des individus.

Les couples d'observations (x_i, y_i) avec $i = 1, \ldots, n$ ordonnées suivant les valeurs croissantes de x sont notés $(x_{(i)}, y_{(i)})$. Nous avons représenté la neuvième valeur de x et sa valeur ajustée $\hat{y}_{(9)} = \hat{\beta}_1 + \hat{\beta}_2 x_{(9)}$ sur le graphique, ainsi que le résidu correspondant $\hat{\varepsilon}_{(9)}$.

1.5.2 Représentation des variables

Nous pouvons voir le problème d'une autre façon. Nous mesurons n couples de points (x_i, y_i). La variable X et la variable Y peuvent être considérées comme deux vecteurs possédant n coordonnées. Le vecteur X (respectivement Y) admet pour coordonnées les observations $x_1, x_2, \ldots, x_n$ (respectivement $y_1, y_2, \ldots, y_n$). Ces deux vecteurs d'observations appartiennent au même espace $\mathbb{R}^n$: l'espace des variables. Nous pouvons donc représenter les données dans l'espace des variables. Le vecteur $\mathbb{1}$ est également un vecteur de $\mathbb{R}^n$ dont toutes les composantes valent 1. Les 2 vecteurs $\mathbb{1}$ et X engendrent un sous-espace de $\mathbb{R}^n$ de dimension 2. Nous avons supposé que $\mathbb{1}$ et X ne sont pas colinéaires grâce à $\mathcal{H}_1$ mais ces vecteurs ne sont pas obligatoirement orthogonaux. Ces vecteurs sont orthogonaux lorsque $\bar{x}$, la moyenne des observations $x_1, x_2, \ldots, x_n$ vaut zéro.
La régression linéaire peut être vue comme la projection orthogonale du vecteur Y dans le sous-espace de $\mathbb{R}^n$ engendré par $\mathbb{1}$ et X, noté $\Im(X)$ (voir fig. 1.11).

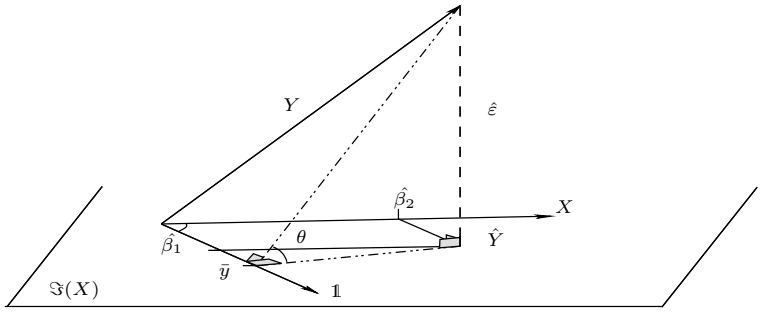


Fig. 1.11 – Représentation de la projection dans l'espace des variables.

Les coefficients $\hat{\beta}_1$ et $\hat{\beta}_2$ s'interprètent comme les composantes de la projection orthogonale notée $\hat{Y}$ de Y sur ce sous-espace.

Remarque
Les vecteurs $\mathbb{1}$ et X de normes respectives $\sqrt{n}$ et $\sqrt{\sum_{i=1}^n x_i^2}$ ne forment pas une base orthogonale. Afin de savoir si ces vecteurs sont orthogonaux, calculons leur produit scalaire. Le produit scalaire est la somme du produit terme à terme des composantes des deux vecteurs et vaut ici $\sum_{i=1}^n x_i \times 1 = n\bar{x}$. Les vecteurs forment une base orthogonale lorsque la moyenne de X est nulle. En effet $\bar{x}$ vaut alors zéro et le produit scalaire est nul. Les vecteurs n'étant en général pas orthogonaux, cela veut dire que $\hat{\beta}_1\mathbb{1}$ n'est pas la projection de Y sur la droite engendrée par $\mathbb{1}$ et que $\hat{\beta}_2 X$ n'est pas la projection de Y sur la droite engendrée par X. Nous reviendrons sur cette différence au chapitre suivant.

Un modèle, que l'on qualifiera de bon, possédera des estimations $\hat{y}_i$ proches des vraies valeurs y_i. Sur la représentation dans l'espace des variables (fig. 1.11) la qualité peut être évaluée par l'angle θ. Cet angle est compris entre -90 degrés et 90 degrés. Un angle proche de -90 degrés ou de 90 degrés indique un modèle de mauvaise qualité. Le cosinus carré de θ est donc une mesure possible de la qualité du modèle et cette mesure varie entre 0 et 1.
Le théorème de Pythagore nous donne directement que

$$\begin{aligned}
\|Y - \bar{y}\mathbb{1}\|^2 &= \|\hat{Y} - \bar{y}\mathbb{1}\|^2 + \|\hat{\varepsilon}\|^2 \\
\sum_{i=1}^n (y_i - \bar{y})^2 &= \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n \hat{\varepsilon}_i^2 \\
\mathrm{SCT} &= \mathrm{SCE} + \mathrm{SCR},
\end{aligned}$$

où SCT (respectivement SCE et SCR) représente la somme des carrés totale (respectivement expliquée par le modèle et résiduelle).

Le coefficient de détermination R^2 est défini par

$$\mathrm{R}^2 = \frac{\mathrm{SCE}}{\mathrm{SCT}} = \frac{\|\hat{Y} - \bar{y}\mathbb{1}\|^2}{\|Y - \bar{y}\mathbb{1}\|^2},$$

c'est-à-dire la part de la variabilité expliquée par le modèle sur la variabilité totale. De nombreux logiciels multiplient cette valeur par 100 afin de donner un pourcentage.

Remarques
Dans ce cas précis, R^2 est le carré du coefficient de corrélation empirique entre les x_i et les y_i et correspond au cosinus carré de l'angle θ :
— si $\mathrm{R}^2 = 1$, le modèle explique tout, l'angle θ vaut donc zéro, Y est dans $\Im(X)$ c'est-à-dire que $y_i = \beta_1 + \beta_2 x_i$;
— si $\mathrm{R}^2 = 0$, cela veut dire que $\sum(\hat{y}_i - \bar{y})^2 = 0$ et donc que $\hat{y}_i = \bar{y}$. Le modèle de régression linéaire est inadapté ;

— si R^2 est proche de zéro, cela veut dire que Y est quasiment dans l'orthogonal de $\Im(X)$, le modèle de régression linéaire est inadapté, la variable X utilisée n'explique pas la variable Y.

1.6 Inférence statistique

Jusqu'à présent, nous avons pu, en choisissant une fonction de coût quadratique, ajuster un modèle de régression, à savoir calculer $\widehat{\beta}_1$ et $\widehat{\beta}_2$. Grâce aux coefficients estimés, nous pouvons donc prédire, pour chaque nouvelle valeur x_{n+1} une valeur de la variable à expliquer $\widehat{y}^p_{n+1}$ qui est tout simplement le point sur la droite ajustée correspondant à l'abscisse x_{n+1}. En ajoutant l'hypothèse $\mathcal{H}_2$, nous avons pu calculer l'espérance et la variance des estimateurs. Ces propriétés permettent d'appréhender de manière grossière la qualité des estimateurs proposés. Le théorème de Gauss-Markov permet de juger de la qualité des estimateurs parmi une classe d'estimateurs : les estimateurs linéaires sans biais. Enfin ces deux hypothèses nous ont aussi permis de calculer l'espérance et la variance de la valeur prédite $\widehat{y}^p_{n+1}$.
Cependant, nous souhaitons en général connaître la loi des estimateurs afin de calculer des intervalles ou des régions de confiance ou effectuer des tests. Il faut donc introduire une hypothèse supplémentaire concernant la loi des ε_i. L'hypothèse $\mathcal{H}_2$ devient

$$\mathcal{H}_3 \left\{ \begin{array}{ll} \varepsilon_i & \sim \mathcal{N}(0, \sigma^2) \\ \varepsilon_i & \text{sont indépendants} \end{array} \right.$$

où $\mathcal{N}(0, \sigma^2)$ est une loi normale d'espérance nulle et de variance σ^2. Le modèle de régression devient le modèle paramétrique $\left\{\mathbb{R}^n, \mathcal{B}_{\mathbb{R}^n}, \mathcal{N}(\beta_1 + \beta_2 x, \sigma^2)\right\}$, où β_1, β_2, σ^2 sont à valeurs dans $\mathbb{R}$, $\mathbb{R}$ et $\mathbb{R}^+$ respectivement. La loi des ε_i étant connue, nous en déduisons la loi des y_i. Toutes les preuves de cette section seront détaillées au chapitre 5.
Nous allons envisager dans cette section les propriétés supplémentaires des estimateurs qui découlent de l'hypothèse $\mathcal{H}_3$ (normalité et indépendance des erreurs) :
— lois des estimateurs $\hat{\beta}_1$, $\hat{\beta}_2$ et $\hat{\sigma}^2$;
— intervalles de confiance univariés et bivariés ;
— loi des valeurs prévues $\hat{y}^p_{n+1}$ et intervalle de confiance.
Cette partie est plus technique que les parties précédentes. Afin de faciliter la lecture, considérons les notations suivantes :

$$\begin{array}{rcl} \sigma^2_{\hat{\beta}_1} & = & \sigma^2 \left(\dfrac{\sum x_i^2}{n \sum (x_i - \bar{x})^2} \right), \qquad \hat{\sigma}^2_{\hat{\beta}_1} = \hat{\sigma}^2 \left(\dfrac{\sum x_i^2}{n \sum (x_i - \bar{x})^2} \right), \\ \sigma^2_{\hat{\beta}_2} & = & \dfrac{\sigma^2}{\sum (x_i - \bar{x})^2}, \qquad \hat{\sigma}^2_{\hat{\beta}_2} = \dfrac{\hat{\sigma}^2}{\sum (x_i - \bar{x})^2}, \end{array}$$

où $\hat{\sigma}^2 = \sum \hat{\varepsilon}_i^2 / (n-2)$. Cet estimateur est donné au théorème 1.4. Notons que les estimateurs de la colonne de gauche ne sont pas réellement des estimateurs.

En effet puisque σ^2 est inconnu, ces estimateurs ne sont pas calculables avec les données. Cependant, ce sont eux qui interviennent dans les lois des estimateurs $\hat{\beta}_1$ et $\hat{\beta}_2$ (voir proposition 1.7). Les estimateurs donnés dans la colonne de droite sont ceux qui sont utilisés (et utilisables) et ils consistent simplement à remplacer σ^2 par $\hat{\sigma}^2$. Les lois des estimateurs sont données dans la proposition suivante.

Proposition 1.7 (Lois des estimateurs : variance connue)
Les lois des estimateurs des MC sont :
(i) $\hat{\beta}_j \sim \mathcal{N}\left(\beta_j, \sigma^2_{\hat{\beta}_j}\right)$ *pour* $j = 1, 2$.
(ii) $\hat{\beta} = \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} \sim \mathcal{N}\left(\beta, \sigma^2 V\right)$, $\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$ *et* $V = \dfrac{1}{\sum(x_i - \bar{x})^2} \begin{bmatrix} \sum x_i^2/n & -\bar{x} \\ -\bar{x} & 1 \end{bmatrix}$.
(iii) $\dfrac{(n-2)}{\sigma^2}\hat{\sigma}^2$ *suit une loi du* χ^2 *à* $(n-2)$ *degrés de liberté (ddl)* (χ^2_{n-2}).
(iv) $(\hat{\beta}_1, \hat{\beta}_2)$ *et* $\hat{\sigma}^2$ *sont indépendants.*

La variance σ^2 n'est pas connue en général, nous l'estimons par $\hat{\sigma}^2$. Les estimateurs des MC ont alors les propriétés suivantes :

Proposition 1.8 (Lois des estimateurs : variance estimée)
Lorsque σ^2 *est estimée par* $\hat{\sigma}^2$, *nous avons*
(i) pour $j = 1, 2$ $\dfrac{\hat{\beta}_j - \beta_j}{\hat{\sigma}_{\hat{\beta}_j}} \sim \mathcal{T}_{n-2}$ *où* $\mathcal{T}_{n-2}$ *est une loi de Student à* $(n-2)$ *ddl.*
(ii) $\dfrac{1}{2\,\hat{\sigma}^2}(\hat{\beta} - \beta)' V^{-1} (\hat{\beta} - \beta) \sim \mathcal{F}_{2,n-2}$, *où* $\mathcal{F}_{2,n-2}$ *est une loi de Fisher à 2 ddl au numérateur et* $(n-2)$ *ddl au dénominateur.*

Ces dernières propriétés nous permettent de donner des intervalles de confiance (IC) ou des régions de confiance (RC) des paramètres inconnus. En effet, la valeur ponctuelle d'un estimateur est en général insuffisante et il est nécessaire de lui adjoindre un intervalle de confiance. Nous parlerons d'IC quand un paramètre est univarié et de RC quand le paramètre est multivarié.

Proposition 1.9 (IC et RC de niveau $1-\alpha$ pour les paramètres)
(i) Un IC *bilatéral de* β_j $(j \in \{1, 2\})$ *est donné par :*

$$\left[\hat{\beta}_j - t_{n-2}(1-\alpha/2)\hat{\sigma}_{\hat{\beta}_j}, \hat{\beta}_j + t_{n-2}(1-\alpha/2)\hat{\sigma}_{\hat{\beta}_j}\right] \tag{1.5}$$

où $t_{n-2}(1-\alpha/2)$ *est le fractile de niveau* $(1-\alpha/2)$ *d'une loi* $\mathcal{T}_{n-2}$.
(ii) Une RC des deux paramètres inconnus β *est donnée par l'équation suivante :*

$$\frac{1}{2\hat{\sigma}^2}\left[n(\hat{\beta}_1 - \beta_1)^2 + 2n\bar{x}(\hat{\beta}_1 - \beta_1)(\hat{\beta}_2 - \beta_2) + \sum x_i^2(\hat{\beta}_2 - \beta_2)^2\right] \leq f_{(2,n-2)}(1-\alpha),$$

$f_{(2,n-2)}(1-\alpha)$ *est le fractile de niveau* $(1-\alpha)$ *d'une loi de Fisher à* $(2, n-2)$ *ddl.*

(iii) Un IC *de* σ^2 *est donné par :*

$$\left[\frac{(n-2)\hat{\sigma}^2}{c_{n-2}(1-\alpha/2)}, \frac{(n-2)\hat{\sigma}^2}{c_{n-2}(\alpha/2)}\right],$$

où $c_{n-2}(1-\alpha/2)$ *représente le fractile de niveau* $(1-\alpha/2)$ *d'une loi du* χ^2 *à* $(n-2)$ *degrés de liberté.*

Remarque
La propriété (ii) donne la RC simultanée des paramètres de la régression $\beta = (\beta_1, \beta_2)'$, appelée ellipse de confiance grâce à la loi du couple. Au contraire (i) donne l'IC d'un paramètre sans tenir compte de la corrélation entre $\hat{\beta}_1$ et $\hat{\beta}_2$. Il est donc délicat de donner une RC du vecteur (β_1, β_2) en juxtaposant les deux IC.

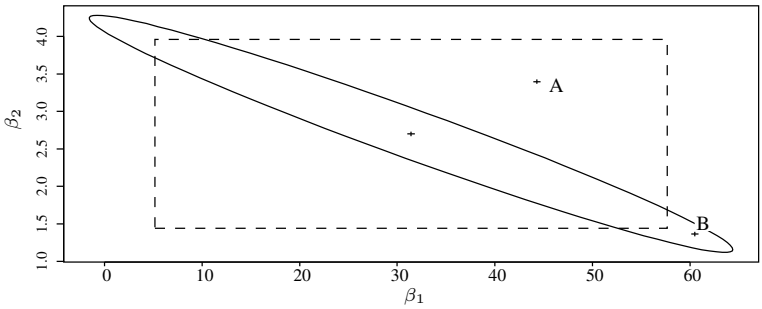


Fig. 1.12 – Comparaison entre ellipse et rectangle de confiance.

Un point peut avoir chaque coordonnée dans son IC respectif mais ne pas appartenir à l'ellipse de confiance. Le point A est un exemple de ce type de point. *A contrario*, un point peut appartenir à la RC sans qu'aucune de ses coordonnées n'appartienne à son IC respectif (le point B). L'ellipse de confiance n'est pas toujours calculée par les logiciels de statistique. Le rectangle de confiance obtenu en juxtaposant les deux intervalles de confiance peut être une bonne approximation de l'ellipse si la corrélation entre $\hat{\beta}_1$ et $\hat{\beta}_2$ est faible(fig. 1.12).

Nous pouvons donner un intervalle de confiance de la droite de régression.

Proposition 1.10 (IC pour $\mathbb{E}(y_i)$)
Un IC *de* $\mathbb{E}(y_i) = \beta_1 + \beta_2 x_i$ *est donné par :*

$$\left[\hat{y}_i \pm t_{n-2}(1-\alpha/2)\hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum(x_l - \bar{x})^2}}\right]. \tag{1.6}$$

En calculant les IC pour tous les points de la droite, nous obtenons une hyperbole de confiance. Lorsque x_i est proche de $\bar{x}$, le terme dominant de la variance est $1/n$, et dès que x_i s'éloigne de $\bar{x}$, le terme dominant est le terme au carré. Nous avons les mêmes résultats que ceux obtenus à la section (1.4.3). Enonçons le résultat permettant de calculer un intervalle de confiance pour une valeur prévue :

Proposition 1.11 (IC pour y_{n+1})
Un IC *de* y_{n+1} *est donné par :*

$$\left[\hat{y}^p_{n+1} \pm t_{n-2}(1-\alpha/2)\hat{\sigma}\sqrt{1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum(x_i - \bar{x})^2}}\right]. \tag{1.7}$$

Plus le point à prévoir est éloigné de $\bar{x}$, plus la variance de la prévision et donc l'IC seront grands. Intuitivement, plus une observation est éloignée du centre de gravité, moins nous avons d'information sur elle. Lorsque x_{n+1} est à l'intérieur de l'étendue des x_i, le terme dominant de la variance est la valeur 1 et donc la variance est relativement constante. Lorsque x_{n+1} est en dehors de l'étendue des x_i, le terme dominant peut être le terme au carré, et la forme de l'intervalle sera à nouveau une hyperbole.

1.7 Exemples

La concentration en ozone

Nous allons traiter les 50 données journalières de concentration en ozone. La variable à expliquer est la concentration en ozone notée `O3` et la variable explicative est la température notée `T12`.

- Nous commençons par représenter les données.

```
> ozone <- read.table("ozone_simple.txt", header = T, sep = ";")
> plot(O3~T12, data=ozone, xlab="T12", ylab="O3")
```

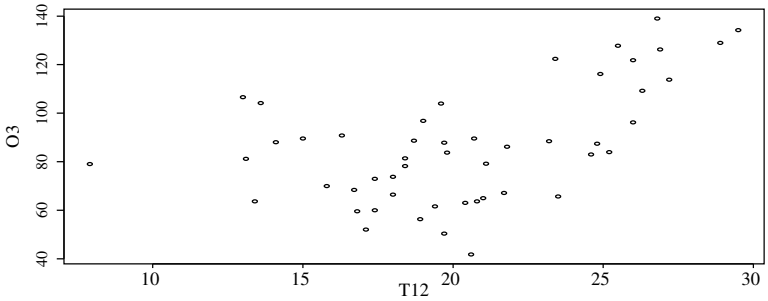


Fig. 1.13 – 50 données journalières de `T12` et `O3`.

Ce graphique permet de vérifier visuellement si une régression linéaire est pertinente. Autrement dit, il suffit de regarder si le nuage de points s'étire le long d'une droite. Bien qu'ici il semble que le nuage s'étire sur une première droite jusqu'à 22 ou 23 degrés C puis selon une autre droite pour les hautes valeurs de températures, nous pouvons tenter une régression linéaire simple.

- Nous effectuons ensuite la régression linéaire, c'est-à-dire la phase d'estimation.

```
> reg <- lm(O3~T12, data=ozone)
```

et analysons les résultats

```
> summary(reg)
Call:
lm(formula = O3 ~ T12)
Residuals:
    Min      1Q  Median      3Q     Max
-45.256 -15.326  -3.461  17.634  40.072

Coefficients
              Estimate    Std. Error    t value    Pr(>|t|)
(Intercept)    31.4150      13.0584      2.406     0.0200       *
T12             2.7010       0.6266      4.311     8.04e-05   ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Residual standard error: 20.5 on 48 degrees of freedom
Multiple R-Squared: 0.2791,     Adjusted R-squared: 0.2641
F-statistic: 18.58 on 1 and 48 DF,  p-value: 8.041e-05
```

Les sorties du logiciel donnent une matrice (sous le mot `Coefficients`) qui comporte pour chaque paramètre (chaque ligne) 5 colonnes. La première colonne contient les estimations des paramètres (colonne `Estimate`), la seconde les écarts-types estimés des paramètres (`Std. Error`). Dans la troisième colonne (`t value`) figure la valeur observée de la statistique de test d'hypothèse $\mathrm{H}_0 : \beta_i = 0$ contre $\mathrm{H}_1 : \beta_i \neq 0$. La quatrième colonne (`Pr(>|t|)`) contient la probabilité critique (ou « p-value ») qui est la probabilité, pour la statistique de test sous H_0, de dépasser la valeur estimée. Enfin la dernière colonne est une version graphique du test : `***` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.001 ; `**` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.01 ; `*` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.05, `.` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.1.
Nous rejetons l'hypothèse H_0 pour les deux paramètres estimés au niveau $\alpha = 5\ \%$. Dans le cadre de la régression simple, cela permet d'effectuer de manière rapide un choix de variable pertinente. En toute rigueur, si pour les deux paramètres l'hypothèse H_0 est acceptée, il est nécessaire de reprendre un modèle en supprimant le paramètre dont la probabilité critique est la plus proche de 1. Dans ce cas-là, dès la phase de représentation des données, de gros doutes doivent apparaître sur l'intérêt de la régression linéaire simple.
Le résumé de l'étape d'estimation fait figurer l'estimation de σ qui vaut ici 20.5 ainsi que le nombre $n - 2 = 48$ qui est le nombre de degrés de liberté associés, par exemple, aux tests d'hypothèse $\mathrm{H}_0 : \beta_i = 0$ contre $\mathrm{H}_1 : \beta_i \neq 0$.
La valeur du R^2 est également donnée, ainsi que le R^2 ajusté noté $\mathrm{R}^2_{\mathrm{a}}$ (voir définition 2.5 p. 43). La valeur du R^2 est faible ($\mathrm{R}^2 = 0.28$) et nous retrouvons la remarque effectuée à propos de la figure (fig. 1.13) : une régression linéaire simple n'est peut-être pas adaptée ici.
La dernière ligne, surtout utile en régression multiple, indique le test entre le mo-

dèle utilisé et le modèle n'utilisant que la constante comme variable explicative. Nous reviendrons sur ce test au chapitre 5.

• Afin d'examiner la qualité du modèle et des observations, nous traçons la droite ajustée et les observations. Comme il existe une incertitude dans les estimations, nous traçons aussi un intervalle de confiance de la droite (à 95 %).

```
> plot(O3~T12, data=ozone)
> T12 <- seq(min(ozone[,"T12"]), max(ozone[,"T12"]), length = 100)
> grille <- data.frame(T12)
> ICdte <- predict(reg, new=grille, interval="conf", level=0.95)
> matlines(grille$T12, cbind(ICdte), lty = c(1,2,2), col = 1)
```

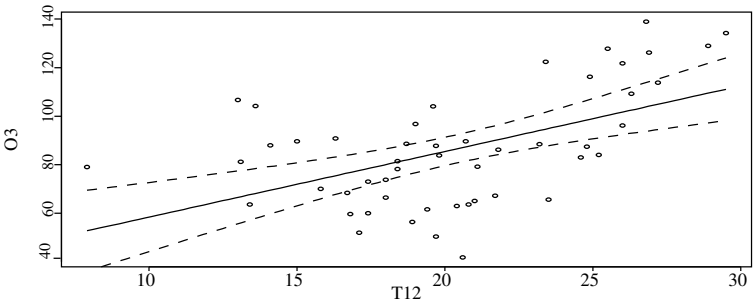


Fig. 1.14 – 50 données journalières de T12 et O3 et l'ajustement linéaire obtenu.

Ce graphique permet de vérifier visuellement si la régression est correcte, c'est-à-dire d'analyser la qualité d'ajustement du modèle. Nous constatons que les observations qui possèdent de faibles valeurs ou de fortes valeurs de température sont au-dessus de la droite ajustée (fig. 1.14) alors que les observations qui possèdent des valeurs moyennes sont en dessous. Les erreurs ne semblent donc pas identiquement distribuées. Pour s'en assurer il serait possible de tracer les résidus. Enfin l'intervalle de confiance à 95 % est éloigné de la droite. Cet intervalle peut être vu comme « le modèle peut être n'importe quelle droite dans cette bande ». Il en découle que la qualité de l'estimation ne semble pas être très bonne.

• Dans une optique de prévision, il est nécessaire de s'intéresser à la qualité de prévision. Cette qualité peut être envisagée de manière succincte grâce à l'intervalle de confiance des prévisions. Afin de bien le distinguer de celui de la droite, nous figurons les deux sur le même graphique (fig. 1.15).

```
> plot(O3~T12, data = ozone, ylim = c(0,150))
> T12 <- seq(min(ozone[,"T12"]), max(ozone[,"T12"]), length = 100)
> grille <- data.frame(T12)
```

```
> ICdte <- predict(reg, new=grille, interval="conf", level=0.95)
> ICprev <- predict(reg, new=grille, interval="pred", level=0.95)
> matlines(T12, cbind(ICdte,ICprev[,-1]), lty=c(1,2,2,3,3), col=1)
> legend("topleft", lty=2:3, c("Y","E(Y)"))
```

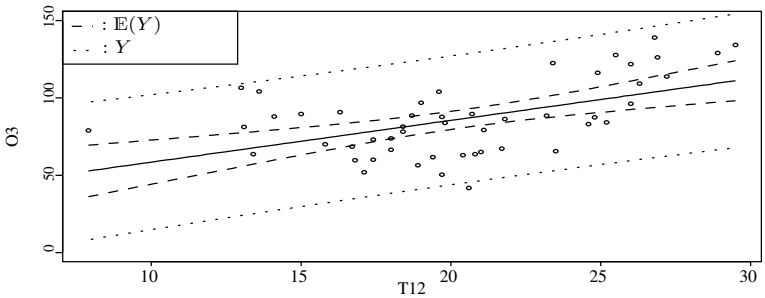


Fig. 1.15 – Droite de régression et intervalles de confiance pour Y et pour $\mathbb{E}(Y)$.

Afin d'illustrer les équations des intervalles de confiance pour les prévisions et la droite ajustée (équations (1.6) et (1.7), p. 21), nous remarquons bien évidemment que l'intervalle de confiance des prévisions est plus grand que l'intervalle de confiance de la droite de régression. L'intervalle de confiance de la droite de régression admet une forme hyperbolique.

- Si nous nous intéressons au rôle des variables, nous pouvons calculer les intervalles de confiance des paramètres *via* la fonction `confint`. Par défaut, le niveau est fixé à 95 %.

```
> IC <- confint(reg, level = 0.95)
> IC
              2.5 %   97.5 %
(Intercept) 5.159232 57.67071
T12         1.441180  3.96089
```

L'IC à 95 % sur l'ordonnée à l'origine est étendu (52.5). Cela provient des erreurs (l'estimateur de σ vaut 20.5), mais surtout du fait que les températures sont en moyenne très loin de 0. Cependant, ce coefficient ne fait pas très souvent l'objet d'interprétation. L'autre IC à 95 % est moins étendu (2.5). Nous constatons qu'il semble exister un effet de la température sur les pics d'ozone.

- Il est conseillé de tracer la région de confiance simultanée des deux paramètres et de comparer cette région aux intervalles de confiance obtenus avec le même degré de confiance. Cette comparaison illustre uniquement la différence entre intervalle simple et région de confiance. En général, l'utilisateur de la méthode choisit l'une ou l'autre forme. Pour cette comparaison, nous utilisons les commandes suivantes :

```
> library(ellipse)
> plot(ellipse(reg, level=0.95), type = "l", xlab = "", ylab = "")
> points(coef(reg)[1], coef(reg)[2], pch = 3)
> lines(IC[1,c(1,1,2,2,1)], IC[2,c(1,2,2,1,1)], lty = 2)
```

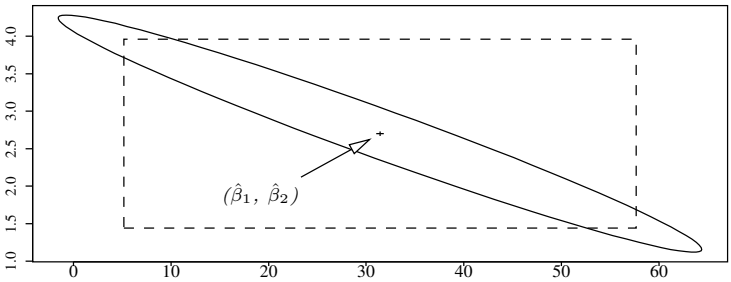


Fig. 1.16 – Région de confiance simultanée des deux paramètres.

Les axes de l'ellipse ne sont pas parallèles aux axes du graphique, les deux estimateurs sont corrélés. Nous retrouvons que la corrélation entre les deux estimateurs est toujours négative (ou nulle), le grand axe de l'ellipse ayant une pente négative. Nous observons bien sûr une différence entre le rectangle de confiance, juxtaposition des deux intervalles de confiance et l'ellipse.

La hauteur des eucalyptus

Nous allons modéliser la hauteur des arbres en fonction de leur circonférence.

- Nous commençons par représenter les données.

```
> eucalypt <- read.table("eucalyptus.txt", header = T, sep = ";")
> plot(ht~circ, data = eucalypt, xlab = "circ", ylab = "ht")
```

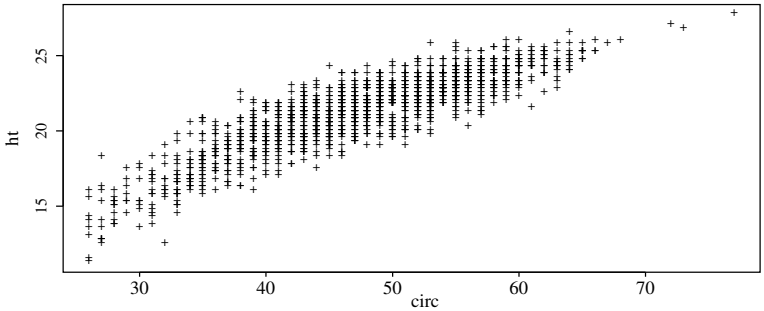


Fig. 1.17 – Représentation des mesures pour les $n = 1429$ eucalyptus mesurés.

Une régression simple semble indiquée, les points étant disposés grossièrement le long d'une droite. Trois arbres ont des circonférences élevées supérieures à 70 cm.

- Nous effectuons ensuite la régression linéaire, c'est-à-dire la phase d'estimation...

```
> reg <- lm(ht~circ, data = eucalypt)
> summary(reg)
Call:
lm(formula = ht ~ circ, data = eucalypt)

Residuals:
     Min       1Q   Median       3Q      Max
-4.76589 -0.78016  0.05567  0.82708  3.69129

Coefficients:
            Estimate  Std. Error  t value  Pr(>|t|)
(Intercept) 9.037476    0.179802    50.26    <2e-16 ***
circ        0.257138    0.003738    68.79    <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.199 on 1427 degrees of freedom
Multiple R-Squared: 0.7683,     Adjusted R-squared: 0.7682
F-statistic:  4732 on 1 and 1427 DF,  p-value: < 2.2e-16
```

Nous retrouvons comme sortie la matrice des informations sur les coefficients, matrice qui comporte 4 colonnes et autant de lignes que de coefficients (voir 1.7, p. 23). Les tests de nullité des deux coefficients indiquent qu'ils semblent tous deux significativement non nuls (quand l'autre coefficient est fixé à la valeur estimée).
Le résumé de l'étape d'estimation fait figurer l'estimation de σ qui vaut ici 1.199 ainsi que le nombre $n - 2 = 1427$ qui est le nombre de degrés de liberté associés, par exemple, aux tests d'hypothèse $\mathrm{H}_0 : \beta_i = 0$ contre $\mathrm{H}_1 : \beta_i \neq 0$.
La valeur du R^2 est également donnée, ainsi que le $\mathrm{R}^2_{\mathrm{a}}$. La valeur du R^2 est élevée ($\mathrm{R}^2 = 0.7683$) et nous retrouvons la remarque déjà faite (fig. 1.17) : une régression linéaire simple semble adaptée.
Le test F entre le modèle utilisé et le modèle n'utilisant que la constante comme variable explicative indique que la circonférence est explicative et que l'on repousse le modèle n'utilisant que la constante comme variable explicative au profit du modèle de régression simple. Ce test n'est pas très utile ici car il équivaut au test de nullité $\mathrm{H}_0 : \beta_2 = 0$ contre $\mathrm{H}_1 : \beta_2 \neq 0$. De plus, dès la première étape, nous avions remarqué que les points s'étiraient le long d'une droite dont le coefficient directeur était loin d'être nul.

- Afin d'examiner la qualité du modèle et des observations, nous traçons la droite ajustée et les observations. Comme il existe une incertitude dans les estimations,

nous traçons aussi un intervalle de confiance de la droite (à 95 %).

```
> plot(ht~circ, data = eucalypt, pch = "+", col = "grey60")
> grille <- data.frame(circ = seq(min(eucalypt[,"circ"]),
+                       max(eucalypt[,"circ"]), length = 100))
> ICdte <- predict(reg, new=grille, interval="confi", level=0.95)
> matlines(grille$circ, ICdte, lty = c(1,2,2), col = 1)
```

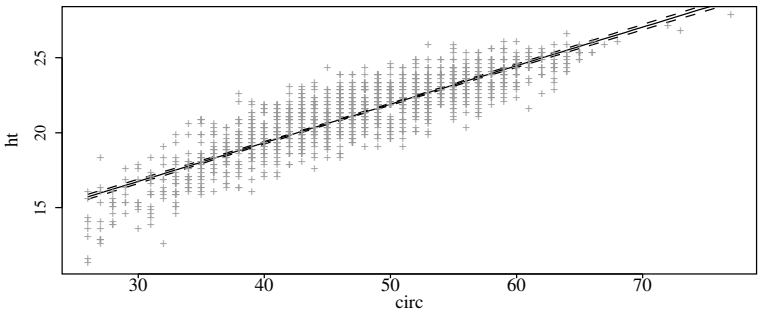


Fig. 1.18 – Données de circonférence/hauteur et ajustement linéaire obtenu.

La figure (1.18) permet de vérifier visuellement si une régression est correcte, c'est-à-dire de constater la qualité d'ajustement de notre modèle. Nous constatons que les observations sont globalement bien ajustées par le modèle, mais les faibles valeurs de circonférences semblent en majorité situées en dessous de la courbe. Ceci indique qu'un remplacement de cette droite par une courbe serait une amélioration possible. Peut-être qu'un modèle de régression simple du type

$$\texttt{ht} \quad = \quad \beta_0 + \beta_1 \sqrt{\texttt{circ}} + \varepsilon,$$

serait plus adapté. Remarquons aussi que les 3 circonférences les plus fortes (supérieures à 70 cm) sont bien ajustées par le modèle. Ces 3 individus sont donc différents en termes de circonférence mais bien ajustés par le modèle.

Enfin, l'intervalle de confiance à 95 % est proche de la droite. Cet intervalle peut être vu comme « le modèle peut être n'importe quelle droite dans cette bande ». Il en découle que la qualité de l'estimation semble être très bonne, ce qui est normal car le nombre d'individus (*i.e.* le nombre d'arbres) est très élevé et les données sont bien réparties le long d'une droite.

- Dans une optique de prévision, il est nécessaire de s'intéresser à la qualité de prévision. Cette qualité peut être envisagée de manière succincte grâce aux intervalles de confiance, de la droite ajustée et des prévisions.

```
> plot(ht~circ, data = eucalypt, pch = "+", col = "grey60")
> circ <- seq(min(eucalypt[,"circ"]),
+                      max(eucalypt[,"circ"]), len = 100)
> grille <- data.frame(circ)
```

```
> ICdte <- predict(reg, new=grille, interval="conf", level=0.95)
> ICprev <- predict(reg, new=grille, interval="pred", level=0.95)
> matlines(circ, cbind(ICdte,ICprev[,-1]),lty=c(1,2,2,3,3), col=1)
```

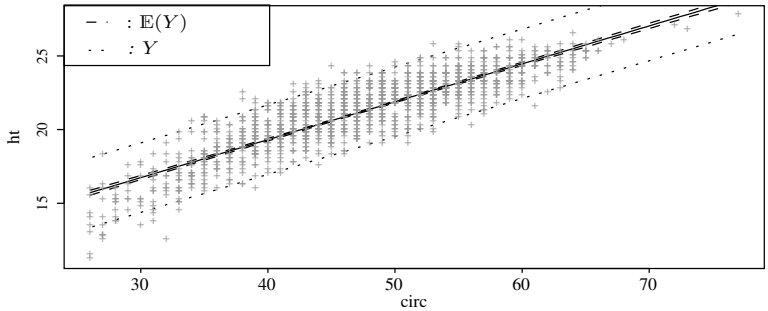


Fig. 1.19 – Droite de régression et intervalles de confiance pour Y et pour $\mathbb{E}(Y)$.

Rien de notable sur l'intervalle de prévision, mis à part le fait qu'il est nécessaire de bien distinguer l'intervalle de confiance de la droite et de la prévision.

1.8 Exercices

Exercice 1.1 (Questions de cours)

1) Lors d'une régression simple, si le R^2 vaut 1, les points sont alignés :
   A. non,
   B. oui,
   C. pas obligatoirement.
2) La droite des MC d'une régression simple passe par le point $(\bar{x}, \bar{y})$:
   A. toujours,
   B. jamais,
   C. parfois.
3) Nous avons effectué une régression simple et recevons une nouvelle observation x_N. Nous calculons la prévision correspondante $\hat{y}_N$, sa variance est minimale lorsque
   A. $x_N = 0$,
   B. $x_N = \bar{x}$,
   C. aucun rapport.
4) Le vecteur $\hat{Y}$ est orthogonal au vecteur des résidus estimés $\hat{\varepsilon}$:
   A. toujours,
   B. jamais,
   C. parfois.

Exercice 1.2 (Biais des estimateurs)
Calculer le biais de $\hat{\beta}_2$ et $\hat{\beta}_1$.

Exercice 1.3 (Variance des estimateurs)
Calculer la variance de $\hat{\beta}_2$ puis la variance de $\hat{\beta}_1$ (indice : calculer $\mathrm{Cov}(\bar{y}, \hat{\beta}_2)$).

Exercice 1.4 (Covariance de $\hat{\beta}_1$ et $\hat{\beta}_2$)
Calculer la covariance entre $\hat{\beta}_1$ et $\hat{\beta}_2$.

Exercice 1.5 (†Théorème de Gauss-Markov)
Démontrer le théorème de Gauss-Markov en posant $\tilde{\beta}_2 = \sum_{i=1}^n \lambda_i y_i$, un estimateur linéaire quelconque (indice : trouver deux conditions sur la somme des λ_i pour que $\tilde{\beta}_2$ ne soit pas biaisé, puis calculer la variance en introduisant $\hat{\beta}_2$).

Exercice 1.6 (Somme des résidus)
Montrer que, dans un modèle de régression linéaire simple, la somme des résidus est nulle.

Exercice 1.7 (Estimateur de la variance du bruit)
Montrer que, dans un modèle de régression linéaire simple, la statistique $\hat{\sigma}^2 = \sum_{i=1}^n \hat{\varepsilon}_i^2/(n-2)$ est un estimateur sans biais de σ^2.

Exercice 1.8 (Prévision)
Calculer la variance de $\hat{y}_{n+1}^p$ puis celle de l'erreur de prévision $\hat{\varepsilon}_{n+1}^p$.

Exercice 1.9 (R^2 et coefficient de corrélation)
Montrer que le R^2 est égal au carré du coefficient de corrélation empirique entre les x_i et y_i.

Exercice 1.10 (Les arbres)
Nous souhaitons exprimer la hauteur y d'un arbre d'une essence donnée en fonction de son diamètre x à 1 m 30 du sol. Pour ce faire, nous avons mesuré 20 couples « diamètre-hauteur ». Nous avons effectué les calculs suivants :

$$\bar{x} = 34.9 \qquad \frac{1}{20}\sum_{i=1}^{20}(x_i - \bar{x})^2 = 28.29 \qquad \bar{y} = 18.34$$

$$\frac{1}{20}\sum_{i=1}^{20}(y_i - \bar{y})^2 = 2.85 \qquad \frac{1}{20}\sum_{i=1}^{20}(x_i - \bar{x})(y_i - \bar{y}) = 6.26.$$

1) On note $\hat{y} = \hat{\beta}_1 + \hat{\beta}_2 x$, la droite de régression. Donner l'expression de $\hat{\beta}_2$ en fonction des statistiques élémentaires ci-dessus. Calculer $\hat{\beta}_1$ et $\hat{\beta}_2$.
2) Donner et commenter une mesure de la qualité de l'ajustement des données au modèle. Exprimer cette mesure en fonction des statistiques élémentaires.
3) Cette question traite des tests qui seront vus au chapitre 5. Cependant, cette question peut être résolue grâce à la section exemple. Les estimations des écarts-types de $\hat{\beta}_1$ et de $\hat{\beta}_2$ donnent $\hat{\sigma}_{\hat{\beta}_1} = 1.89$ et $\hat{\sigma}_{\hat{\beta}_2} = 0.05$. Testez $H_0 : \beta_j = 0$ contre $H_1 : \beta_j \neq 0$ pour $j = 0,\ 1$. Pourquoi ce test est-il intéressant dans notre contexte ? Que pensez-vous du résultat ?

Exercice 1.11 (Modèle quadratique)
Au vu du graphique 1.13, nous souhaitons modéliser l'ozone par la température au carré.
1) Ecrire le modèle et estimer les paramètres.
2) Comparer ce modèle au modèle linéaire classique.

Chapitre 2

La régression linéaire multiple

2.1 Introduction

La modélisation de la concentration d'ozone dans l'atmosphère évoquée au chapitre 1 est relativement simpliste. En effet, des variables météorologiques autres que la température peuvent expliquer cette concentration, par exemple le rayonnement, la précipitation ou encore le vent qui déplace les masses d'air. L'association Air Breizh mesure ainsi en même temps que la concentration d'ozone les variables météorologiques susceptibles d'avoir une influence sur celle-ci. Voici quelques-unes de ces données :

Individu	`O3`	`T12`	`Vx`	`Ne12`
1	63.6	13.4	9.35	7
2	89.6	15	5.4	4
3	79	7.9	19.3	8
4	81.2	13.1	12.6	7
5	88	14.1	-20.3	6

Tableau 2.1 – 5 données journalières.

La variable `Vx` est une variable synthétique représentant le vent. Le vent est normalement mesuré en degré (direction) et mètre par seconde (vitesse). La variable créée est la projection du vent sur l'axe est-ouest, elle tient compte de la direction et de la vitesse. La variable `Ne12` représente la nébulosité mesurée à 12 heures.
Pour analyser la relation entre la température (`T12`), le vent (`Vx`), la nébulosité à midi (`Ne12`) et l'ozone (`O3`), nous allons chercher une fonction f telle que

$$\texttt{O3}_i \approx f(\texttt{T12}_i, \texttt{Vx}_i, \texttt{Ne12}_i).$$

Afin de préciser le sens de $\approx$, il faut définir un critère positif quantifiant la qualité de l'ajustement de la fonction f aux données. Cette notion de coût permet d'appréhender de manière aisée les problèmes d'ajustement économique dans certains modèles. Minimiser un coût nécessite la connaissance de l'espace sur lequel on minimise, donc la classe de fonctions $\mathcal{G}$ dans laquelle nous supposerons que se trouve la vraie fonction inconnue.
Le problème mathématique peut s'écrire de la façon suivante :

$$\underset{f\in\mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} l(y_i - f(x_{i1}, \cdots, x_{ip})),$$

où n représente le nombre de données à analyser et $l(.)$ est appelée fonction de coût. La fonction de coût sera la même que celle utilisée précédemment, c'est-à-dire le coût quadratique. En ce qui concerne le choix de la classe $\mathcal{G}$, nous utiliserons pour commencer la classe des fonctions linéaires :
$\mathcal{G}\left\{f : f(x_1, \cdots, x_p) = \sum_{j=1}^{p} \beta_j x_j \quad \text{avec} \quad \beta_j \in \mathbb{R}, j \in \{1, \ldots, p\}\right\}$.

2.2 Modélisation

Le modèle de régression multiple est une généralisation du modèle de régression simple lorsque les variables explicatives sont en nombre fini. Nous supposons donc que les données collectées suivent le modèle suivant :

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i, \qquad i = 1, \cdots, n \tag{2.1}$$

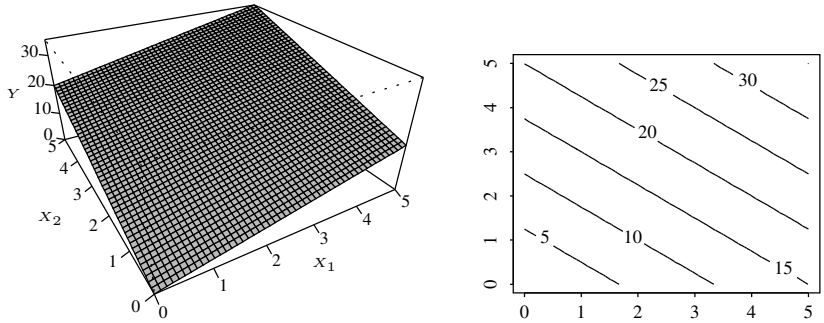
où
— les x_{ij} sont des nombres connus, non aléatoires. La variable x_{i1} peut valoir 1 pour tout i variant de 1 à n. Dans ce cas, β_1 représente la constante (`intercept` dans les logiciels anglo-saxons). En statistiques, cette colonne de 1 est presque toujours présente ;
— les paramètres à estimer β_j du modèle sont inconnus ;
— les ε_i sont des variables aléatoires inconnues.
En utilisant l'écriture matricielle de (2.1), nous obtenons la définition suivante :

Définition 2.1 (Modèle de régression multiple)
Un modèle de régression linéaire est défini par une équation de la forme

$$Y_{n\times 1} = X_{n\times p}\ \beta_{p\times 1} + \varepsilon_{n\times 1}. \tag{2.2}$$

où :
• *Y est un vecteur aléatoire de dimension n ;*
• *X est une matrice de taille $n \times p$ connue, appelée matrice du plan d'expérience, X est la concaténation des p variables X_j : $X = (X_1|X_2|\ldots|X_p)$. Nous noterons la i^e ligne du tableau X par le vecteur ligne $x_i' = (x_{i1}, \ldots, x_{ip})$;*
• *β est le vecteur de dimension p des paramètres inconnus du modèle ;*
• *ε est le vecteur centré, de dimension n, des erreurs.*

Nous supposons que la matrice X est de plein rang. Cette hypothèse sera notée $\mathcal{H}_1$. Comme, en général, le nombre d'individus n est plus grand que le nombre de variables explicatives p, le rang de la matrice X vaut p.
La présentation précédente revient à supposer que la fonction liant Y aux variables explicatives X est un hyperplan représenté ci-dessous (fig. 2.1).



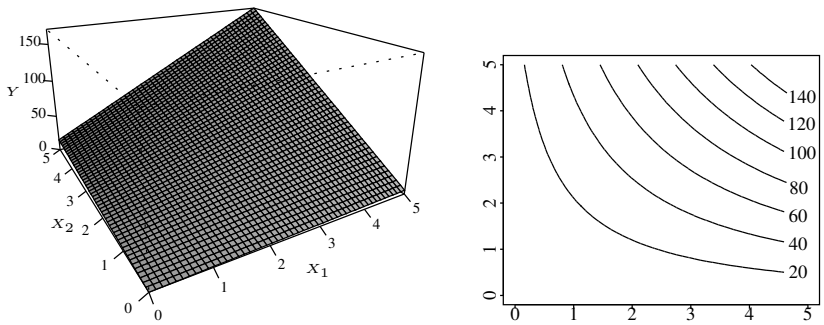
(a) Surface. **(b)** Contour.

Fig. 2.1 – Représentation géométrique de la relation $Y = 3X_1 + 4X_2$.

Il est naturel dans nombre de problèmes de penser que des interactions existent entre les variables explicatives. Dans l'exemple de l'ozone, nous pouvons penser que la température et le vent interagissent. Pour modéliser cette interaction, nous écrivons en général un modèle avec un produit entre les variables explicatives qui interagissent. Ainsi, pour deux variables, nous avons la modélisation suivante :

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i1} x_{i2} + \varepsilon_i, \qquad i = 1, \cdots, n.$$

Les produits peuvent s'effectuer entre deux variables définissant des interactions d'ordre 2, entre trois variables définissant des interactions d'ordre 3, etc. D'un point de vue géométrique, cela donne (fig. 2.2) :



(a) Surface. **(b)** Contour.

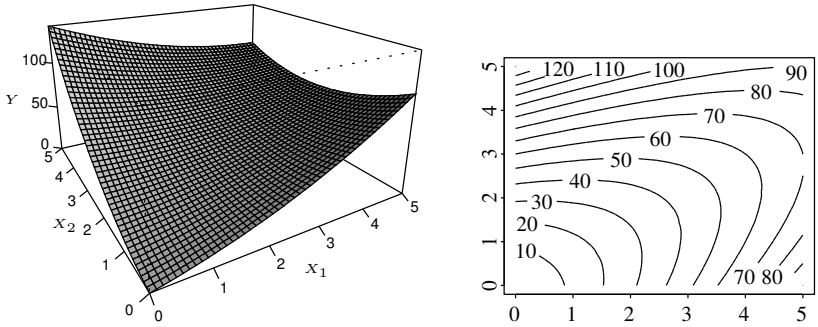
Fig. 2.2 – Représentation géométrique de la relation $Y = X_1 + 3X_2 + 6X_1X_2$.

Cependant, ce type de modélisation rentre parfaitement dans le cadre de la régression multiple. Les variables d'interaction sont des produits de variables connues et

sont donc connues. Dans l'exemple précédent, la troisième variable explicative X_3 sera tout simplement le produit X_1X_2 et nous retrouvons la modélisation proposée à la section précédente.
De même, d'autres extensions peuvent être utilisées comme le modèle de régression polynomial. En reprenant notre exemple à deux variables explicatives X_1 et X_2, nous pouvons proposer le modèle polynomial de degré 2 suivant :

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i1} x_{i2} + \beta_4 x_{i1}^2 + \beta_5 x_{i2}^2 + \varepsilon_i, \qquad i = 1, \cdots, n.$$

Ce modèle peut être remis dans la formulation de la section précédente en posant $X_3 = X_1X_2$, $X_4 = X_1^2$ et $X_5 = X_2^2$. L'hypersurface ressemble alors à (fig. 2.3) :



(a) Surface. **(b)** Contour.

Fig. 2.3 – Représentation de la relation $Y = 10X_1 + 8X_2 - 6X_1X_2 + 2X_1^2 + 4X_2^2$.

En conclusion, nous pouvons considérer que n'importe quelle transformation connue et fixée des variables explicatives (logarithme, exponentielle, produit, etc.) rentre dans le modèle de régression multiple. Ainsi la transformée d'une variable explicative X_1 par la fonction log par exemple devient $\tilde{X}_1 = \log(X_1)$ et le modèle reste donc un modèle de régression multiple. Par contre une transformation comme $\exp\{-r(X_1 - k)\}$ qui est une fonction non linéaire de deux paramètres inconnus r et k ne rentre pas dans ce cadre. En effet, ne connaissant pas r et k, il est impossible de calculer $\exp\{-r(X_1 - k)\}$ et donc de la noter $\tilde{X}_1$. Ce type de relation est traité dans Antoniadis *et al.* (1992). Ainsi un modèle linéaire ne veut pas forcément dire que le lien entre variables explicatives et la variable à expliquer est linéaire mais que le *modèle est linéaire en les paramètres* β_j.

2.3 Estimateurs des moindres carrés

Définition 2.2 (Estimateur des MC)
On appelle estimateur des moindres carrés (noté MC) $\hat{\beta}$ *de* β *la valeur suivante :*

$$\hat{\beta} = \underset{\beta_1, \cdots, \beta_p}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|Y - X\beta\|^2.$$

Théorème 2.1 (Expression de l'estimateur des MC)
Si l'hypothèse $\mathcal{H}_1$ est vérifiée, l'estimateur des MC $\hat{\beta}$ de β vaut

$$\hat{\beta} = (X'X)^{-1}X'Y.$$

La section suivante est entièrement consacrée à ce résultat.

2.3.1 Calcul de $\hat{\beta}$

Il est intéressant de considérer les vecteurs dans l'espace des variables ($\mathbb{R}^n$)(fig. 2.4). Ainsi, Y, vecteur colonne, définit dans $\mathbb{R}^n$ un vecteur $\overrightarrow{OY}$ d'origine O et d'extrémité Y. Ce vecteur a pour coordonnées $(y_1, \cdots, y_n)$. La matrice X du plan d'expérience est formée de p vecteurs colonnes. Chaque vecteur X_j définit dans $\mathbb{R}^n$ un vecteur $\overrightarrow{OX_j}$ d'origine O et d'extrémité X_j. Ce vecteur a pour coordonnées $(x_{1j}, \cdots, x_{nj})$. Ces p vecteurs linéairement indépendants (hypothèse $\mathcal{H}_1$) engendrent un sous-espace vectoriel de $\mathbb{R}^n$, noté dorénavant $\Im(X)$, de dimension p.

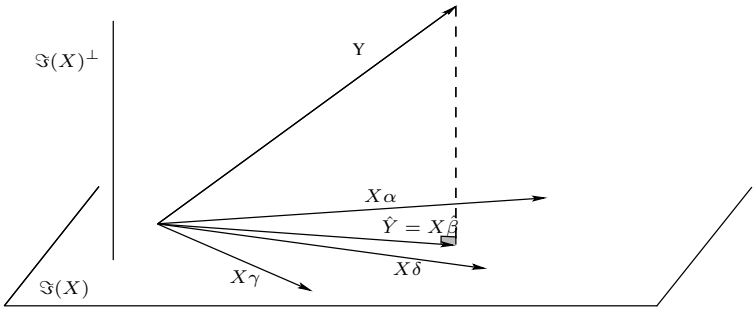


Fig. 2.4 – Représentation dans l'espace des variables.

Cet espace $\Im(X)$, appelé image de X (voir annexe A), est engendré par les colonnes de X. Il est parfois appelé espace des solutions. L'espace orthogonal à $\Im(X)$, noté $\Im(X)^{\perp}$, est souvent appelé espace des résidus. Tout vecteur $\overrightarrow{v}$ de $\Im(X)$ s'écrit de façon unique sous la forme suivante :

$$\overrightarrow{v} = \alpha_1\overrightarrow{X_1} + \cdots + \alpha_p\overrightarrow{X_p} = X\alpha,$$

où $\alpha = [\alpha_1, \cdots, \alpha_p]'$. Selon le modèle (2.2), le vecteur Y est la somme d'un élément de $\Im(X)$ et d'un bruit, élément de $\mathbb{R}^n$, qui n'a aucune raison d'appartenir à $\Im(X)$. Minimiser $S(\beta)$ revient à chercher un élément de $\Im(X)$ qui soit le plus proche de Y, au sens de la norme euclidienne classique. Par définition, cet unique élément noté $\hat{Y}$ est appelé projection orthogonale de Y sur $\Im(X)$ noté

$$\hat{Y} \quad = \quad P_X Y = X\hat{\beta}.$$

La matrice P_X est la matrice de projection orthogonale sur $\Im(X)$ et $\hat{\beta}$ est l'estimateur des moindres carrés de β. Le vecteur $\hat{Y}$ contient les valeurs ajustées de Y par le modèle.

Dans la littérature anglo-saxonne, cette matrice est souvent notée H et est appelée « hat matrix » car elle met des « hat » sur Y. Par souci de cohérence de l'écriture, nous noterons h_{ij} l'élément courant (i, j) de P_X.

• Calcul de $\hat{\beta}$ par projection :
Trois possibilités de calcul de $\hat{\beta}$ sont proposées.
— La première consiste à connaître la forme analytique de P_X. La matrice de projection orthogonale sur $\Im(X)$ est donnée par :

$$P_X = X(X'X)^{-1}X'$$

et, comme $P_X Y = X\hat{\beta}$, nous obtenons $\hat{\beta} = (X'X)^{-1}X'Y$.

— La deuxième méthode utilise le fait que le vecteur Y de $\mathbb{R}^n$ se décompose de façon unique en une partie sur $\Im(X)$ et une partie sur $\Im(X)^{\perp}$, cela s'écrit :

$$Y = P_X Y + (I - P_X)Y.$$

La quantité $(I-P_X)Y$ étant un élément de $\Im(X)^{\perp}$ est orthogonale à tout élément v de $\Im(X)$. Rappelons que $\Im(X)$ est l'espace engendré par les colonnes de X, c'est-à-dire que toutes les combinaisons linéaires de variables $X_1, \cdots, X_p$ sont éléments de $\Im(X)$ ou encore que, pour tout $\alpha \in \mathbb{R}^p$, nous avons $X\alpha \in \Im(X)$. Les deux vecteurs v et $(I - P_X)Y$ étant orthogonaux, le produit scalaire entre ces deux quantités est nul, soit :

$$\begin{aligned}
\langle v, (I - P_X)Y\rangle &= 0 \quad \forall v \in \Im(X) \\
\langle X\alpha, (I - P_X)Y\rangle &= 0 \quad \forall \alpha \in \mathbb{R}^p \\
\alpha' X'(I - P_X)Y &= 0 \\
X'Y &= X'P_X Y \quad \text{avec} \quad P_X Y = X\hat{\beta} \\
X'Y &= X'X\hat{\beta} \qquad X \text{ de rang plein} \\
\hat{\beta} &= (X'X)^{-1}X'Y.
\end{aligned}$$

Nous retrouvons $P_X = X(X'X)^{-1}X'$, matrice de projection orthogonale sur l'espace engendré par les colonnes de X. Les propriétés caractéristiques d'un projecteur orthogonal ($P'_X = P_X$ et $P_X^2 = P_X$) sont vérifiées.

— La dernière façon de procéder consiste à écrire que le vecteur $(I - P_X)Y$ est orthogonal à chacune des colonnes de X :

$$\left\{\begin{array}{lcl}
\langle X_1, Y - X\hat{\beta}\rangle & = & 0 \\
\vdots & & \\
\langle X_p, Y - X\hat{\beta}\rangle & = & 0
\end{array}\right. \quad \Leftrightarrow \quad X'Y = X'X\hat{\beta}.$$

Soit $P_X = X(X'X)^{-1}X'$ la matrice de projection orthogonale sur $\Im(X)$, la matrice de projection orthogonale sur $\Im(X)^{\perp}$ est $P_{X^{\perp}} = (I - P_X)$.

• Calcul matriciel
Nous pouvons aussi retrouver le résultat précédent de manière analytique en écrivant la fonction à minimiser $S(\beta)$:

$$\begin{aligned} S(\beta) &= \|Y - X\beta\|^2 \\ &= Y'Y + \beta'X'X\beta - Y'X\beta - \beta'X'Y \\ &= Y'Y + \beta'X'X\beta - 2Y'X\beta. \end{aligned}$$

Une condition nécessaire d'optimum est que la dérivée première par rapport à β s'annule. Or la dérivée s'écrit comme suit (voir annexe A) :

$$\frac{\partial S(\beta)}{\partial \beta} = -2X'Y + 2X'X\beta,$$

d'où, s'il existe, l'optimum, noté $\hat{\beta}$, vérifie

$$-2X'Y + 2X'X\hat{\beta} = 0$$

c'est-à-dire $\hat{\beta} = (X'X)^{-1}X'Y$.
Pour s'assurer que ce point $\hat{\beta}$ est bien un minimum strict, il faut que la dérivée seconde soit une matrice définie positive. Or la dérivée seconde s'écrit

$$\frac{\partial^2 S(\beta)}{\partial \beta^2} = 2X'X,$$

et X est de plein rang donc $X'X$ est inversible et n'a pas de valeur propre nulle. La matrice $X'X$ est donc définie. De plus $\forall z \in \mathbb{R}^p$, nous avons

$$z'2X'Xz = 2\langle Xz, Xz\rangle = 2\|Xz\|^2 \geq 0$$

$(X'X)$ est donc bien définie positive et $\hat{\beta}$ est bien un minimum strict.

2.3.2 Interprétation

Nous venons de voir que $\hat{Y}$ est la projection de Y sur le sous-espace engendré par les colonnes de X. Cette projection existe et est unique même si l'hypothèse $\mathcal{H}_1$ n'est pas vérifiée. L'hypothèse $\mathcal{H}_1$ nous permet en fait d'obtenir un $\hat{\beta}$ unique. Dans ce cas, s'intéresser aux coordonnées de $\hat{\beta}$ a un sens, et ces coordonnées sont les coordonnées de $\hat{Y}$ dans le repère $X_1, \cdots, X_p$. Ce repère n'a aucune raison d'être orthogonal et donc $\hat{\beta}_j$ n'est pas la coordonnée de la projection de Y sur X_j. Dans ce cas (usuel en pratique), effectuer des régressions simples pour estimer chaque β_j ne donnera bien évidemment pas les mêmes résultats qu'effectuer une régression multiple. En effet, nous avons

$$P_X Y = \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p.$$

Calculons la projection de Y sur X_j.

$$\begin{aligned} P_{X_j}Y &= P_{X_j}P_X Y \\ &= \hat{\beta}_1 P_{X_j}X_1 + \cdots + \hat{\beta}_p P_{X_j}X_p \\ &= \hat{\beta}_j X_j + \sum_{i\neq j} \hat{\beta}_i P_{X_j}X_i. \end{aligned}$$

Cette dernière quantité est différente de $\hat{\beta}_j X_j$ sauf si X_j est orthogonal à toutes les autres variables. L'exercice 2.8 illustre ces différences.
Lorsque toutes les variables sont orthogonales deux à deux, il est clair que $(X'X)$ est une matrice diagonale

$$(X'X) = \mathrm{diag}(\|X_1\|^2, \cdots, \|X_p\|^2). \tag{2.3}$$

2.3.3 Quelques propriétés statistiques

Le statisticien cherche à vérifier que les estimateurs des MC que nous avons construits admettent de bonnes propriétés au sens statistique. Dans notre cadre de travail, cela peut se résumer en deux parties : l'estimateur des MC est-il sans biais et est-il de variance minimale dans sa classe d'estimateurs ?
Pour cela, nous supposons une seconde hypothèse, notée $\mathcal{H}_2$, indiquant que les erreurs sont centrées, de même variance (homoscédasticité) et non corrélées entre elles. L'écriture de cette hypothèse est $\mathcal{H}_2 : \mathbb{E}(\varepsilon) = 0, \quad \Sigma_\varepsilon = \sigma^2\, I_n$, avec I_n la matrice identité d'ordre n. Cette hypothèse nous permet de calculer

$$\mathbb{E}(\hat{\beta}) = \mathbb{E}((X'X)^{-1}X'Y) = (X'X)^{-1}X'\mathbb{E}(Y) = (X'X)^{-1}X'X\beta = \beta.$$

L'estimateur des MC est donc sans biais. Calculons sa variance

$$\mathrm{V}(\hat{\beta}) = \mathrm{V}((X'X)^{-1}X'Y) = (X'X)^{-1}X'\,\mathrm{V}(Y)X(X'X)^{-1} = \sigma^2(X'X)^{-1}.$$

Proposition 2.1 ($\hat{\beta}$ sans biais)
L'estimateur $\hat{\beta}$ des MC est un estimateur sans biais de β et sa variance vaut $\mathrm{V}(\hat{\beta}) = \sigma^2(X'X)^{-1}$.

Remarque
Lorsque les variables sont orthogonales deux à deux, les composantes de $\hat{\beta}$ ne sont pas corrélées entre elles puisque la matrice $(X'X)$ est diagonale (2.3).

Le théorème de Gauss-Markov (voir exercice 2.3) indique que parmi tous les estimateurs linéaires sans biais de β, l'estimateur obtenu par MC admet la plus petite variance :

Théorème 2.2 (Gauss-Markov)
L'estimateur des MC est optimal parmi les estimateurs linéaires sans biais de β.

En général, le biais et la variance d'un estimateur sont des propriétés qui évoluent en sens contraire et un critère souvent utilisé en statistique est l'erreur quadratique moyenne (EQM).

Définition 2.3 (Erreur Quadratique Moyenne)
L'erreur quadratique moyenne (EQM) d'un estimateur $\hat{\theta}$ de θ de dimension p est

$$\begin{aligned}\mathrm{EQM}(\hat{\theta}) &= \mathbb{E}((\theta-\hat{\theta})(\theta-\hat{\theta})') \\ &= \mathbb{E}(\theta-\hat{\theta})\mathbb{E}(\theta-\hat{\theta})' + V(\hat{\theta}),\end{aligned}$$

c'est-à-dire le biais « au carré » plus la variance.

Un estimateur biaisé peut être meilleur qu'un estimateur non biaisé si sa variance est plus petite. Illustrons, sur un exemple simple, l'équilibre biais variance.
Supposons que nous connaissions la valeur du paramètre θ, ici $\theta = 0$ ainsi que la loi de deux estimateurs $\hat{\theta}_1$ et $\hat{\theta}_2$: $\hat{\theta}_1 \sim \mathcal{N}(-0.5, 1)$ et $\hat{\theta}_2 \sim \mathcal{N}(0, 3^2)$. Nous savons donc que $\hat{\theta}_1$ est biaisé, car $\mathbb{E}(\hat{\theta}_1) = -0.5 \neq \theta$ mais pas $\hat{\theta}_2$. *A priori*, nous serions tentés de prendre $\hat{\theta}_2$, puisqu'en moyenne il « tombe » sur le vrai paramètre θ. Comparons plus attentivement ces deux estimateurs en traçant leur densité. La figure 2.5 présente les densités de ces deux estimateurs et un intervalle de confiance à 95 % de ceux-ci. Si nous choisissons $\hat{\theta}_1$, la distance entre le vrai paramètre et une estimation est, en moyenne, plus faible que pour le choix de $\hat{\theta}_2$. La moyenne de cette distance euclidienne peut être calculée, c'est l'EQM. Ici l'EQM de $\hat{\theta}_1$ vaut 1.25 (biais au carré + variance) et celle de $\hat{\theta}_2$ vaut 3 donc le choix de $\hat{\theta}_1$ est plus raisonnable que $\hat{\theta}_2$: en moyenne il ne vaudra pas la valeur du paramètre, il est biaisé, mais en général il « tombe » moins loin du paramètre car il est moins variable (faible variance).

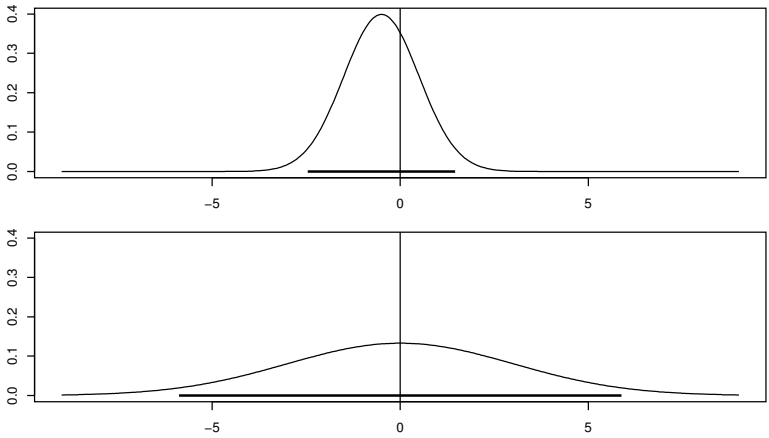


Fig. 2.5 – Estimateurs biaisé et non biaisé. En trait plein est représentée la densité de l'estimateur biaisé (en haut) et non biaisé (en bas). La droite verticale représente la valeur du paramètre à estimer. Le segment horizontal épais représente l'étendue correspondant à 95 % de la probabilité.

L'EQM permet donc de comparer les estimateurs d'un même paramètre. Il est le résultat d'un équilibre entre le biais et la variance, qui réagissent en général en sens contraire. Dans le cas présent, l'estimateur $\hat{\beta}$ est sans biais donc son EQM correspond à sa variance mais nous allons dans ce livre travailler avec des estimateurs biaisés et l'EQM sera donc à utiliser.

2.3.4 Résidus et variance résiduelle

Les résidus sont définis par la relation suivante :

$$\hat{\varepsilon} \quad = \quad Y - \hat{Y} = (I - P_X)Y = P_{X^\perp}Y.$$

Les résidus appartiennent donc à $\Im(X)^\perp$ et cet espace est aussi appelé espace des résidus. Les résidus sont donc toujours orthogonaux à $\hat{Y}$. Afin de calculer facilement les propriétés des résidus, il est possible d'écrire les résidus sous la forme suivante

$$\hat{\varepsilon} \quad = \quad P_{X^\perp}Y = P_{X^\perp}(X\beta + \varepsilon) = P_{X^\perp}\varepsilon$$

et nous avons les propriétés suivantes (voir exercice 2.2).

Proposition 2.2 (Propriétés de $\hat{\varepsilon}$ et $\hat{Y}$)
Sous les hypothèses $\mathcal{H}_1$ et $\mathcal{H}_2$, nous avons

$$\begin{aligned}
\mathbb{E}(\hat{\varepsilon}) &= P_{X^\perp}\mathbb{E}(\varepsilon) = 0 \quad \textit{et} \quad \mathrm{V}(\hat{\varepsilon}) = \sigma^2 P_{X^\perp} I P'_{X^\perp} = \sigma^2 P_{X^\perp} \\
\mathbb{E}(\hat{Y}) &= X\mathbb{E}(\hat{\beta}) = X\beta \quad \textit{et} \quad \mathrm{V}(\hat{Y}) = \sigma^2 P_X \\
\mathrm{Cov}(\hat{\varepsilon}, \hat{Y}) &= 0.
\end{aligned}$$

Les résidus estimés $\hat{\varepsilon}$ de ε possèdent la même espérance que ε. Nous étudierons les résidus plus en détail au chapitre 3. Nous avons mentionné un estimateur de σ^2 noté $\hat{\sigma}^2$. Un estimateur « naturel » de la variance résiduelle est donné par

$$\frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i^2 = \frac{1}{n}\|\hat{\varepsilon}\|^2.$$

Or comme $\|\hat{\varepsilon}\|^2$ est un scalaire, nous écrivons que ce scalaire est égal à sa trace puis, en nous servant de la propriété de la trace (voir annexe A), nous obtenons

$$\mathbb{E}(\|\hat{\varepsilon}\|^2) = \mathbb{E}[\mathrm{tr}(\hat{\varepsilon}'\hat{\varepsilon})] = \mathbb{E}[\mathrm{tr}(\hat{\varepsilon}\hat{\varepsilon}')] = \mathrm{tr}(\mathbb{E}[\hat{\varepsilon}\hat{\varepsilon}']) = \mathrm{tr}(\sigma^2 P_{X^\perp}) = \sigma^2(n-p).$$

La dernière égalité provient du fait que la trace d'un projecteur est égale à la dimension du sous-espace sur lequel on projette. Cet estimateur « naturel » est biaisé. Afin d'obtenir un estimateur sans biais, nous définissons donc

$$\hat{\sigma}^2 = \frac{\|\hat{\varepsilon}\|^2}{n-p} = \frac{\mathrm{SCR}}{n-p},$$

où SCR est la somme des carrés résiduelle.

Proposition 2.3 ($\hat{\sigma}^2$ sans biais)
La statistique $\hat{\sigma}^2$ est un estimateur sans biais de σ^2.

A partir de cet estimateur, nous obtenons un estimateur de la variance de $\hat{\beta}$:

$$\hat{\sigma}^2_{\hat{\beta}} = \hat{\sigma}^2(X'X)^{-1} = \frac{\text{SCR}}{n-p}(X'X)^{-1},$$

et donc un estimateur de l'écart-type de l'estimateur $\hat{\beta}_j$ de chaque coefficient

$$\hat{\sigma}_{\hat{\beta}_j} = \sqrt{\hat{\sigma}^2[(X'X)^{-1}]_{jj}}.$$

2.3.5 Prévision

Un des buts de la régression est de proposer des prévisions pour la variable à expliquer y lorsque nous avons de nouvelles valeurs de x. Soit une nouvelle valeur $x'_{n+1} = (x_{n+1,1}, \cdots, x_{n+1,p})$, nous voulons prédire y_{n+1}. Or

$$y_{n+1} = x'_{n+1}\beta + \varepsilon_{n+1},$$

avec $\mathbb{E}(\varepsilon_{n+1}) = 0$, $\text{V}(\varepsilon_{n+1}) = \sigma^2$ et $\text{Cov}(\varepsilon_{n+1}, \varepsilon_i) = 0$ pour $i = 1, \cdots, n$. Nous pouvons prédire la valeur correspondante grâce au modèle ajusté

$$\hat{y}^p_{n+1} = x'_{n+1}\hat{\beta}.$$

Deux types d'erreurs vont entacher la prévision, la première due à l'incertitude sur ε_{n+1} et l'autre à l'incertitude due à l'estimation. Calculons la variance de l'erreur de prévision

$$\begin{aligned}\text{V}\left(y_{n+1} - \hat{y}^p_{n+1}\right) &= \text{V}(x'_{n+1}\beta + \varepsilon_{n+1} - x'_{n+1}\hat{\beta}) = \sigma^2 + x'_{n+1}V(\hat{\beta})x_{n+1} \\ &= \sigma^2(1 + x'_{n+1}(X'X)^{-1}x_{n+1}).\end{aligned}$$

Nous retrouvons bien l'incertitude due aux erreurs σ^2 à laquelle vient s'ajouter l'incertitude d'estimation.

Remarque
Puisque l'estimateur $\hat{\beta}$ est un estimateur non biaisé de β et l'espérance de ε vaut zéro, les espérances de y_{n+1} et $\hat{y}^p_{n+1}$ sont identiques. La variance de l'erreur de prévision s'écrit :

$$\text{V}\left(y_{n+1} - \hat{y}^p_{n+1}\right) = \mathbb{E}\left[y_{n+1} - \hat{y}^p_{n+1} - \mathbb{E}(y_{n+1}) + \mathbb{E}(\hat{y}^p_{n+1})\right]^2 = \mathbb{E}(y_{n+1} - \hat{y}^p_{n+1})^2.$$

Nous voyons donc ici que la variance de l'erreur de prévision est mesurée par l'erreur quadratique moyenne de prévision (EQMP), quantité que nous retrouverons au chapitre 7 car elle joue un rôle central dans l'évaluation de la qualité des modèles.

2.4 Interprétation géométrique

Le théorème de Pythagore donne directement l'égalité suivante :

$$\begin{aligned} \|Y\|^2 &= \|\hat{Y}\|^2 + \|\hat{\varepsilon}\|^2 \\ &= \|X\hat{\beta}\|^2 + \|Y - X\hat{\beta}\|^2. \end{aligned}$$

Si la constante fait partie du modèle, alors nous avons toujours par le théorème de Pythagore

$$\begin{aligned} \|Y - \bar{y}\mathbb{1}\|^2 &= \|\hat{Y} - \bar{y}\mathbb{1}\|^2 + \|\hat{\varepsilon}\|^2 \\ \text{SC totale} &= \text{SC expliquée par le modèle} + \text{SC résiduelle} \\ \text{SCT} &= \text{SCE} + \text{SCR}\,. \end{aligned}$$

Définition 2.4 (R^2)
Le coefficient de détermination (multiple) R^2 *est défini par*

$$\mathrm{R}^2 = \frac{\|\hat{Y}\|^2}{\|Y\|^2} = \cos^2\theta_0$$

et si la constante fait partie de $\Im(X)$ *par*

$$\mathrm{R}^2 = \frac{\textit{V. expliquée par le modèle}}{\textit{Variation totale}} = \frac{\|\hat{Y} - \bar{y}\mathbb{1}\|^2}{\|Y - \bar{y}\mathbb{1}\|^2} = \cos^2\theta.$$

Le R^2 peut aussi s'écrire en fonction des résidus (voir l'exercice 5.3) :

$$\mathrm{R}^2 = 1 - \frac{\|\hat{\varepsilon}\|^2}{\|Y - \bar{y}\mathbb{1}\|^2}.$$

Ce coefficient mesure le cosinus carré de l'angle entre les vecteurs Y et $\hat{Y}$ pris à l'origine ou pris en $\bar{y}$ (voir fig. 2.6). Ce dernier est toujours plus grand que le premier, le R^2 calculé lorsque la constante fait partie de $\Im(X)$ est donc plus petit que le R^2 calculé directement.

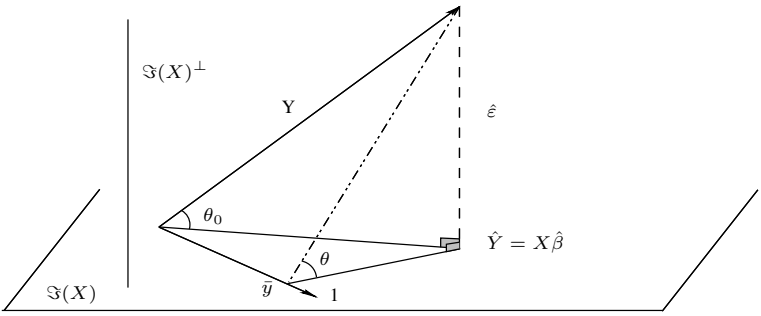


Fig. 2.6 – Représentation des variables et interprétation géométrique du R^2.

Ce coefficient ne tient cependant pas compte de la dimension de $\Im(X)$, un R^2 ajusté est donc défini.

Définition 2.5 (R^2 ajusté)
Le coefficient de détermination ajusté $\mathrm{R}_{\mathrm{a}}^2$ *est défini par*

$$\mathrm{R}_{\mathrm{a}}^2 = 1 - \frac{n}{n-p}\frac{\|\hat{\varepsilon}\|^2}{\|Y\|^2}$$

et, si la constante fait partie de $\Im(X)$*, par*

$$\mathrm{R}_{\mathrm{a}}^2 = 1 - \frac{n-1}{n-p}\frac{\|\hat{\varepsilon}\|^2}{\|Y - \bar{y}\mathbb{1}\|^2}.$$

L'ajustement correspond à la division des normes au carré par leur degré de liberté (ou dimension du sous-espace auquel le vecteur appartient) respectif.

2.5 Exemples

La concentration en ozone

Nous expliquons l'ozone (`O3`) par deux variables explicatives, la température à 12 h (`T12`) et le vent (`Vx`). Le vent est mesuré en degré (direction) et mètre par seconde (vitesse). Nous avons synthétisé ces deux variables en créant une variable (`Vx`) qui est la projection du vent sur l'axe est-ouest. Nous avons $n = 50$ observations. Nous avons choisi deux variables explicatives afin de pouvoir continuer à représenter directement les données et le modèle. Au-delà de deux variables explicatives, il est impossible de visualiser simplement les données. Nous commençons notre étude, à l'image de la régression simple, en traçant les données.

```
> ozone <- read.table("ozone.txt", header = T, sep = ";")
> library("scatterplot3d")
> scatterplot3d(ozone[,"T12"],ozone[,"Vx"],ozone[,"O3"],type="h",
+       pch=16, box=FALSE, xlab="T12", ylab="Vx", zlab="O3")
```

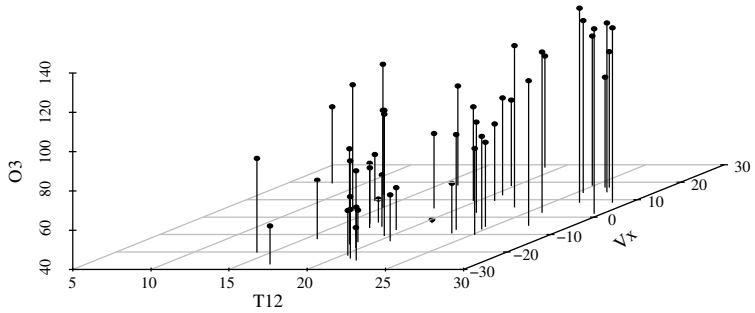


Fig. 2.7 – Représentation brute des données : modèle d'explication de l'ozone (`O3`) par la température à 12 h (`T12`) et le vent (`Vx`).

Il est maintenant très difficile de voir si une régression est adaptée, ce qui signifie ici que les points ne doivent pas être très éloignés d'un plan commun.

- Les phases d'estimation puis de synthèse des résultats obtenus sont conduites avec les ordres suivants :

```
> regmulti <- lm(O3~T12+Vx, data = ozone)
> summary(regmulti)
```

Rappelons que, classiquement, le statisticien inclut toujours une moyenne générale (ou `intercept`). Les logiciels de statistique ne font pas exception à cette règle et ils intègrent automatiquement la moyenne générale, c'est-à-dire la variable X_1 composée uniquement de 1. Le modèle de régression est donc

$$\texttt{O3} \quad = \quad \beta_1 + \beta_2 \texttt{T12} + \beta_3 \texttt{Vx} + \varepsilon$$

Le résumé permet de connaître les estimations des paramètres et de leur écart-type. Il donne aussi la qualité d'ajustement *via* le R^2, ici $\mathrm{R}^2 = 0.52$.

```
Call:
lm(formula = O3 ~ T12 + Vx, data = ozone)

Residuals:
    Min      1Q  Median      3Q     Max
-42.984 -10.152  -2.407  11.710  34.494

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  35.4530    10.7446   3.300  0.00185 **
T12           2.5380     0.5151   4.927 1.08e-05 ***
Vx            0.8736     0.1772   4.931 1.06e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.82 on 47 degrees of freedom
Multiple R-Squared: 0.5249,     Adjusted R-squared: 0.5047
F-statistic: 25.96 on 2 and 47 DF,  p-value: 2.541e-08
```

L'estimation de $\hat{\sigma}$ vaut ici 16.82 et nous avons $n = 50$ pour $p = 3$ variables, ce qui donne $n - p = 47$ (degrés de liberté).
Enfin, à l'issue de cette phase d'estimation, nous pouvons tracer le plan d'équation $z = 35.453 + 2.538x + 0.8736y$. Il est difficile d'avoir une idée de la qualité d'ajustement du modèle *via* une figure en 3 dimensions. En général la qualité d'un modèle sera envisagée par l'analyse des résidus (chapitre 3).

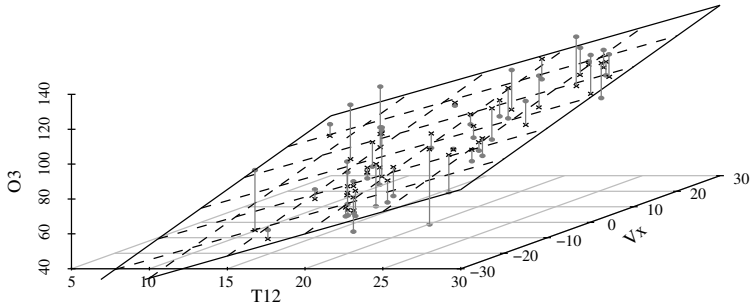


Fig. 2.8 – Représentation des données et hyperplan.

Nous avons ajouté la variable `Vx` au modèle présenté dans le chapitre 1, cet ajout est-il pertinent ? Afin de répondre à cette question nous devons envisager de construire soit des procédures générales de choix de modèles (voir chapitre 7), soit un test entre le modèle de la régression simple $\texttt{O3} = \beta_1 + \beta_2\texttt{T12} + \varepsilon$ et le modèle $\texttt{O3} = \beta_1 + \beta_2\texttt{T12} + \beta_3\texttt{Vx} + \varepsilon$, ce qui est un des objets du chapitre 5.

La hauteur des eucalyptus

Nous cherchons à expliquer la hauteur de $n = 1429$ eucalyptus par leur circonférence. Nous avions mentionné dans le chapitre précédent qu'un modèle du type

$$\texttt{ht} \quad = \quad \beta_1 + \beta_2\texttt{circ} + \beta_3\sqrt{\texttt{circ}} + \varepsilon,$$

serait peut-être plus adapté.

- Nous commençons par représenter les données.

```
> eucalypt <- read.table("eucalyptus.txt", header = T, sep = ";")
> plot(ht~circ, data = eucalypt, xlab = "circ", ylab = "ht")
```

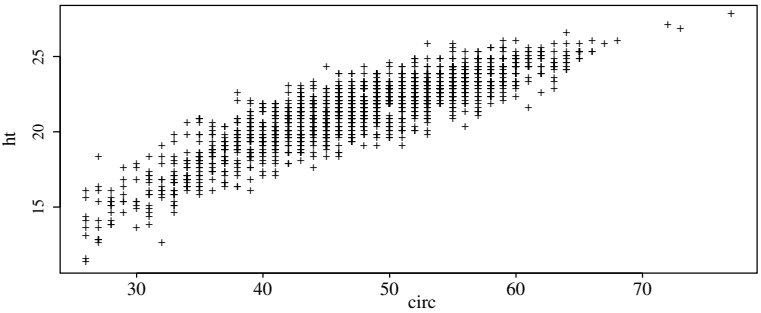


Fig. 2.9 – Représentation des mesures pour les $n = 1429$ eucalyptus mesurés.

- La phase d'estimation permet d'obtenir le résumé ci-après. Notez que l'opéra-

teur `I()` permet de protéger[1] l'opération « racine carrée ». Bien qu'il ne soit pas obligatoire dans ce cas, il est préférable de s'habituer à son emploi.

```
> regmult <- lm(ht ~ circ + I(sqrt(circ)), data = eucalypt)
> resume.mult <- summary(regmult)
> resume.mult
Call:
lm(formula = ht ~ circ + I(sqrt(circ)), data = eucalypt)

Residuals:
     Min       1Q   Median       3Q      Max
-4.18811 -0.68811  0.04272  0.79272  3.74814

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   -24.35200    2.61444  -9.314   <2e-16 ***
circ           -0.48295    0.05793  -8.336   <2e-16 ***
I(sqrt(circ))   9.98689    0.78033  12.798   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.136 on 1426 degrees of freedom
Multiple R-Squared: 0.7922,     Adjusted R-squared: 0.7919
F-statistic:  2718 on 2 and 1426 DF,  p-value: < 2.2e-16
```

L'estimation des 3 coefficients est donnée dans la première colonne, suivie de leur écart-type estimé et du test de nullité du coefficient (voir prochain chapitre). L'estimation de σ donne ici 1.136, avec $n-p=1426$. Le R^2 augmente avec ce nouveau modèle et passe de 0.768 à 0.792. Cela signifie que le modèle ajuste mieux les données avec une variable supplémentaire ($\sqrt{\texttt{circ}}$). Ce phénomène est normal puisque l'on a projeté sur un sous-espace $\Im(X)$ plus grand (on a ajouté une variable), la projection $\hat{Y}=P_X Y$ est plus proche de Y avec le grand modèle et donc le R^2 est meilleur (voir 7.3 p. 171). Le R^2 n'est donc pas adapté pour juger de la pertinence de l'ajout de variables.

- La qualité d'ajustement peut être envisagée graphiquement grâce à :

```
> plot(ht ~ circ, data = eucalypt, pch = "+", col = "grey60")
> grille <- data.frame(circ = seq(min(eucalypt[,"circ"]),
+                        max(eucalypt[,"circ"]), length = 100)
> lines(grille[,"circ"], predict(regmult, grille))
```

1. Noter que le « + » qui sépare deux variables dans la formule `ht~circ+I(sqrt(circ))` ne signifie pas que l'on additionne les 2 variables `circ` et $\sqrt{\texttt{circ}}$. Les opérateurs classiques (`+`, `*`, `^`) que l'on veut utiliser dans les formules doivent être protégés. Ici l'opérateur $\sqrt{\ }$ est protégé par `I()`.

Nous pouvons constater que le modèle semble très bien ajusté pour la plupart des valeurs de circonférence, sauf pour les grandes valeurs (`circ` > 65 cm) où l'ajustement est toujours plus faible que la valeur mesurée.

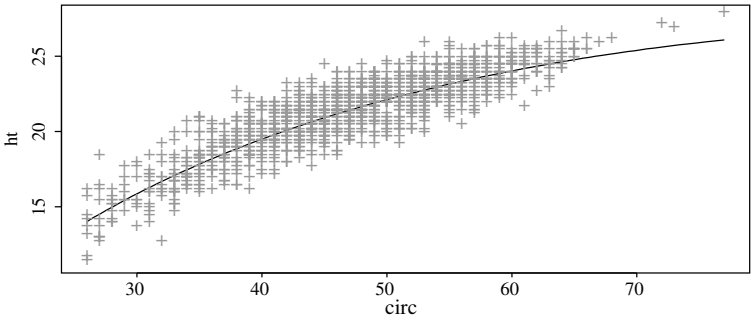


Fig. 2.10 – Représentation des données et du modèle ajusté.

Ce modèle est donc cohérent pour des valeurs jusqu'à 60-65 cm de circonférence et moins adapté au-delà.

2.6 Exercices

Exercice 2.1 (Questions de cours)

1) Nous avons effectué une régression multiple, une des variables explicatives est la constante, la somme des résidus calculés vaut :
   A. 0,
   B. approximativement 0,
   C. parfois 0.

2) Le vecteur $\hat{Y}$ est orthogonal au vecteur des résidus estimés $\hat{\varepsilon}$:
   A. oui,
   B. non,
   C. seulement si $\mathbb{1}$ fait partie des variables explicatives.

3) Un estimateur de la variance de $\hat{\beta}$ de l'estimateur des MC de β vaut :
   A. $\sigma^2(X'X)^{-1}$,
   B. $\hat{\sigma}^2(X'X)^{-1}$,
   C. $\hat{\sigma}^2(XX')^{-1}$.

4) Un autre estimateur $\tilde{\beta}$ que celui des moindres carrés (moindres valeurs absolues par exemple ou d'autres encore) a été calculé. La SCR obtenue avec cet estimateur est :
   A. plus petite que la SCR obtenue avec l'estimateur des MC classique,
   B. plus grande que la SCR obtenue avec l'estimateur des MC classique,
   C. aucun rapport.

5) Une régression a été effectuée et le calcul de la SCR a donné la valeur notée SCR1. Une variable est ajoutée, le calcul de la SCR a donné une nouvelle valeur notée SCR2. Nous savons que :
   A. SCR1 $\leq$ SCR2,
   B. SCR1 $\geq$ SCR2,
   C. cela dépend de la variable ajoutée.

6) Une régression a été effectuée et un estimateur de la variance résiduelle a donné la valeur notée $\hat{\sigma}_1^2$. Une variable est ajoutée et un estimateur de la variance résiduelle vaut maintenant $\hat{\sigma}_2^2$. Nous savons que :
A. $\hat{\sigma}_1^2 \leq \hat{\sigma}_2^2$,
B. $\hat{\sigma}_1^2 \geq \hat{\sigma}_2^2$,
C. on ne peut rien dire.

Exercice 2.2 (Covariance de $\hat{\varepsilon}$ et de $\hat{Y}$)
Montrer que $\mathrm{Cov}(\hat{\varepsilon}, \hat{Y}) = 0$.

Exercice 2.3 (†Théorème de Gauss-Markov)
Démontrer le théorème de Gauss-Markov.

Exercice 2.4 (Représentation des variables)
Nous avons une variable Y à expliquer par une variable X. Nous avons effectué $n = 2$ mesures et trouvé

$$(x_1, y_1) = (4, 5) \quad \text{et} \quad (x_2, y_2) = (1, 5).$$

Représenter les variables, estimer β dans le modèle $y_i = \beta x_i + \varepsilon_i$ puis représenter $\hat{Y}$.
Nous avons maintenant une variable Y à expliquer grâce à 2 variables X et Z, nous avons effectué $n = 3$ mesures

$$(x_1, z_1, y_1) = (3, 2, 0), \quad (x_2, z_2, y_2) = (3, 3, 5) \quad \text{et} \quad (x_3, z_3, y_3) = (0, 0, 3).$$

Représenter les variables, estimer β dans le modèle $y_i = \beta x_i + \gamma z_i + \varepsilon_i$ et représenter $\hat{Y}$.

Exercice 2.5 (Modèles emboîtés)
Soit X une matrice de taille $n \times p$ composée de p vecteurs linéairement indépendants de $\mathbb{R}^n$. Nous notons X_q la matrice composée des q ($q < p$) premiers vecteurs de X. Nous avons les deux modèles suivants :

$$\begin{aligned} Y &= X\beta + \varepsilon \\ Y &= X_q\gamma + \epsilon. \end{aligned}$$

Comparer les R^2 dans les deux modèles.

Exercice 2.6
On examine l'évolution d'une variable Y en fonction de deux variables exogènes x et z. On dispose de n observations de ces variables. On note $X = (\mathbb{1}\ x\ z)$ où $\mathbb{1}$ est le vecteur constant et x, z sont les vecteurs des variables explicatives.
1) Nous avons obtenu les résultats suivants :

$$X'X = \begin{pmatrix} 30 & 0 & 0 \\ ? & 10 & 7 \\ ? & ? & 15 \end{pmatrix}.$$

a) Que vaut n ? Renseigner les valeurs manquantes.
b) Calculer le coefficient de corrélation linéaire empirique entre X et Z.
2) La régression linéaire empirique de Y sur $\mathbb{1}, X, Z$ donne

$$Y = -2\mathbb{1} + X + 2Z + \hat{\varepsilon}, \qquad \mathrm{SCR} = \|\hat{\varepsilon}\|^2 = 12.$$

a) Déterminer la moyenne arithmétique $\bar{y}$.
b) Calculer la somme des carrés expliquée (SCE), la somme des carrés totale (SCT) et le coefficient de détermination.

Exercice 2.7 (Changement d'échelles des variables explicatives)
Nous souhaitons expliquer la variable Y par p variables explicatives $X_1, \cdots, X_p$. L'estimateur des moindres carrés obtenus est $\hat{\beta}$. Pour différentes raisons, chaque variable est pré-multipliée par un scalaire $a_1, \cdots, a_p$. Avec ces nouvelles variables, l'estimateur des moindres carrés obtenus est $\tilde{\beta}$. Montrer que cela revient à diviser chaque composante de $\hat{\beta}$ par les a_i respectifs. Vérifier ce résultat avec R.

Exercice 2.8 (Différence entre régression multiple et régressions simples)
Nous souhaitons expliquer la variable Y par deux variables exogènes X et Z. Nous disposons de n observations de ces variables.
1) Considérons le modèle $y_i = \beta_1 x_i + \beta_2 z_i + \varepsilon_i$, donner la forme explicite de $\hat{\beta}_1$ et $\hat{\beta}_2$.
2) Considérons maintenant les deux modèles univariés $y_i = \beta_1 x_i + \varepsilon_i$ et $y_i = \beta_2 z_i + \varepsilon_i$, donner la forme explicite des estimateurs.
3) Que concluez-vous ?
4) Considérons l'analyse séquentielle : estimer les paramètres du modèle $y_i = \beta_1 x_i + \varepsilon_i$ puis expliquer les résidus par la variable Z. Comparer les estimateurs.

Exercice 2.9 (TP : différence entre régression multiple et régressions simples)
L'objectif de ce TP est d'illustrer l'exercice précédent. Simulez 2 variables explicatives X_1 et X_2 qui suivent par exemple une loi uniforme entre $[0, 1]$ puis simulez ensuite le modèle

$$Y = 2 - 3X_1 + 4X_2 + \varepsilon,$$

où ε suit une loi normale d'espérance 0 et de variance .2.
1) Dessiner le nuage de points en utilisant la fonction **plot3d** du package **rgl**.
2) Qu'observez-vous ?
3) Effectuer une régression multiple, estimer les paramètres, conserver $\hat{Y}$.
4) Effectuer les régressions simples de Y sur X_1 et sur X_2 estimez le paramètre associé (est-ce le même ?).
5) Effectuer maintenant deux régressions séquentielles : régresser Y sur X_1, conserver les résidus et effectuer une régression simple des résidus sur X_2. Comparer les paramètres estimés et comparer $\hat{Y}$ avec le $\hat{Y}$ de la question 3.
6) Changer l'ordre de la question précédente : commencer par régresser Y sur X_2 puis les résidus sur X_1, comparer les estimateurs et $\hat{Y}$.
7) Les variables X_1 et X_2 étaient peu corrélées (utilisé **cov**) mais elles n'étaient pas orthogonales (et oui il faut les centrer !). Recommencer tout l'exercice en remplaçons X_1 par $X_1c = X_1 - mean(X_1)$ et X_2 par $X_2c = X_2 - mean(X_2)$. Conclusion ?

Exercice 2.10 (TP : régression multiple et codes R)
L'objectif de ce TP est d'utiliser R pour effectuer des modèles de régression en utilisant la fonction **formula**. Importer les données d'ozone et effectuez une régression multiple en utilisant la fonction **lm** où la variable à expliquer est `O3` grâce aux variables de température, nébulosité...
Il est inutile d'écrire toutes les variables mais juste `lm(O3~ .,data=vosdonnees)`. Il est parfois utile d'écrire un modèle avec les variables, pour faire cette manipulation,
1) Récupérer les noms des variables explicatives dans un vecteur `nomvar`.

2) A l'aide de la fonction **paste** et de l'argument collapse, constituer le vecteur de caractères suivant `"Ne9+Ne12+···+O3v"`.
3) Modifier cette chaîne de caractères pour obtenir la chaîne suivante :
`"O3 ~ Ne9+Ne12+···+O3v"`.
4) Transformer cette chaîne de caractères en utilisant la fonction **formula** et affecter le résultat dans l'objet `formule`. Vérifier que vous pouvez appliquer la fonction **lm** à cet objet.

Exercice 2.11 (Régression orthogonale)
Considérons le modèle $Y = X\beta + \varepsilon$, où $Y \in \mathbb{R}^n$, X est une matrice de taille $n \times p$ composée de p vecteurs orthogonaux, $\beta \in \mathbb{R}^p$ et $\varepsilon \in \mathbb{R}^n$. Considérons U la matrice des q premières colonnes de X et V la matrice des $p-q$ dernières colonnes de X. Nous avons obtenu par les MC les estimations suivantes :

$$\begin{aligned}
\hat{Y}_X &= \hat{\beta}_1^X X_1 + \cdots + \hat{\beta}_p^X X_p \\
\hat{Y}_U &= \hat{\beta}_1^U X_1 + \cdots + \hat{\beta}_q^U X_q \\
\hat{Y}_V &= \hat{\beta}_{q+1}^V X_{q+1} + \cdots + \hat{\beta}_p^V X_p.
\end{aligned}$$

Notons également $\mathrm{SCE}(A)$ la norme au carré de $P_A Y$.
1) Montrer que $\mathrm{SCE}(X) = \mathrm{SCE}(U) + \mathrm{SCE}(V)$.
2) Choisir une variable nommée X_I, montrer que l'estimation de β_I est identique quel que soit le modèle utilisé.

Exercice 2.12 (Centrage, centrage-réduction et coefficient constant)
Considérons le modèle $Y = X\beta + \varepsilon$ où la dernière colonne (la p^{e}) de X est le vecteur $\mathbb{1}$.
1) Soient les variables $\{X_j\}$, $j = 1, \cdots, p$ et Y et celles centrées notées $\{\tilde{X}_j\}$ et $\tilde{Y}$. Montrer que la dernière colonne de $\tilde{X}$ regroupant les variables $\{\tilde{X}_j\}$ vaut 0. La matrice $\tilde{X}$ sera dorénavant la matrice X centrée et privée de sa dernière colonne de 0. Elle est donc de dimension $n \times (p-1)$.
2) Soit le modèle suivant : $\tilde{Y} = \tilde{X}\tilde{\beta} + \varepsilon$. En identifiant ce modèle avec le modèle de régression $Y = X\beta + \varepsilon$, trouver la valeur de β_p en fonction de $\tilde{\beta}_1, \ldots, \tilde{\beta}_{p-1}$ et des moyennes empiriques de Y et X. Ce coefficient β_p associé à la variable $\mathbb{1}$ est appelé coefficient constant (ou *intercept* en anglais).
3) Supposons maintenant que les variables $\{X_j\}$ sont centrées-réduites et que Y est simplement centrée. Nous continuons à les noter $\{\tilde{X}_j\}$ et $\tilde{Y}$. Que valent $\beta_1, \ldots, \beta_{p-1}$ en fonction de $\tilde{\beta}_1, \ldots, \tilde{\beta}_{p-1}$? Que vaut le coefficient constant β_p ?
4) Même question que précédemment avec X et Y centrées-réduites.
5) Simuler 3 variables explicatives et effectuer les questions précédentes sous R.

Exercice 2.13 (†† Moindres carrés contraints)
Considérons le modèle $Y = X\beta + \varepsilon$. Nous définissons l'estimateur des MC classique et l'estimateur contraint par

$$\begin{aligned}
\hat{\beta} &= \operatorname{argmin} \|Y - X\beta\|^2 \\
\hat{\beta}_c &= \operatorname{argmin} \|Y - X\beta\|^2 \quad \text{sc} \quad R\beta = r,
\end{aligned}$$

où R est une matrice de taille $q \times p$ de rang $q \leq p$ et r un vecteur de $\mathbb{R}^q$.
1) Calculer l'estimateur des moindres carrés.
2) Vérifier que l'estimateur des moindres carrés contraints vaut

$$\hat{\beta}_c = \hat{\beta} + (X'X)^{-1}R'[R(X'X)^{-1}R']^{-1}(r - R\hat{\beta}).$$

3) Calculer l'EQM de $\hat{\beta}_c$ et comparer à l'EQM de l'estimateur des MC.

Chapitre 3

Validation du modèle

Nous rappelons le contexte :

$$Y_{n\times 1} = X_{n\times p}\ \beta_{p\times 1} + \varepsilon_{n\times 1},$$

sous les hypothèses

- $\mathcal{H}_1 : \mathrm{rang}(X) = p$,
- $\mathcal{H}_2 : \mathbb{E}(\varepsilon) = 0, \quad \Sigma_\varepsilon = \sigma^2\, \mathrm{I}_n$ cette hypothèse est souvent remplacée par
- $\mathcal{H}_3 : \varepsilon \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_n)$.

Les différentes étapes d'une régression peuvent se résumer de la sorte :

1. la modélisation : nous avons supposé que la variable Y est expliquée de manière linéaire par les variables $X_1, \cdots, X_p$ *via* le modèle de régression $Y = X\beta + \varepsilon$;

2. l'estimation : nous avons ensuite estimé les paramètres grâce aux données récoltées. Les hypothèses sur le bruit ε notées $\mathcal{H}_2$ ou $\mathcal{H}_3$ ont permis d'établir des propriétés statistiques des estimateurs obtenus ;

3. la validation : objectif de ce chapitre. Nous aborderons le problème de la validation des hypothèses $\mathcal{H}_2$ ou $\mathcal{H}_3$. La vérification de l'hypothèse $\mathcal{H}_1$ est immédiate et les solutions dans le cas où cette hypothèse n'est pas vérifiée seront abordées aux chapitres 4 et 9. Nous envisagerons aussi les problèmes d'ajustement d'un individu ainsi que la validation du modèle lui-même (validation globale), problème important mais souvent négligé. Cette validation globale peut être envisagée de deux manières : choix ou non d'inclure des variables et/ou vérification du caractère linéaire de la liaison entre la variable considérée et Y comme spécifié par le modèle. Nous traiterons ici le caractère linéaire de la liaison et les transformations éventuelles à effectuer pour rendre cette liaison linéaire. Le choix d'inclure ou de retirer des variables sera étudié en détail au chapitre 7.

3.1 Analyse des résidus

L'examen des résidus constitue une étape primordiale de la régression linéaire. Cette étape est essentiellement fondée sur des méthodes graphiques, et il est donc difficile d'avoir des règles strictes de décision. L'objectif de cette partie est de présenter ces méthodes graphiques. Commençons par rappeler les définitions des différents résidus.

3.1.1 Les différents résidus

Les erreurs ε_i sont estimées par $\hat{\varepsilon}_i = y_i - \hat{y}_i$. Nous avons

Hypothèses	Réalité
$\mathbb{E}(\varepsilon_i) = 0$	$\mathbb{E}(\hat{\varepsilon}_i) = 0$
$\mathrm{V}(\varepsilon) = \sigma^2 I$	$\mathrm{V}(\hat{\varepsilon}) = \sigma^2(I - P_X)$

Afin d'éliminer la non-homogénéité des variances des résidus estimés, nous préférons utiliser les résidus normalisés définis par

$$r_i = \frac{\hat{\varepsilon}_i}{\sigma\sqrt{1 - h_{ii}}},$$

où h_{ij} est l'élément (i, j) de la matrice P_X. Nous obtenons les résidus standardisés en remplaçant le paramètre inconnu σ par son estimateur $\hat{\sigma}$:

$$t_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}\sqrt{1 - h_{ii}}}. \tag{3.1}$$

Ces résidus ne sont pas indépendants par construction et ils ne peuvent donc pas être représentatifs d'une absence/présence de structuration par autocorrélation. Leur loi est difficile à calculer car le numérateur et le dénominateur sont corrélés. Ils possèdent la même variance unité, ils sont donc utiles pour détecter des valeurs importantes de résidus. Cependant, nous préférons utiliser les résidus studentisés par validation croisée (VC) (souvent nommés dans les logiciels *studentized residuals*)

$$t_i^* = \frac{\hat{\varepsilon}_i}{\hat{\sigma}_{(i)}\sqrt{1 - h_{ii}}},$$

où $\hat{\sigma}_{(i)}$ est l'estimateur de σ dans le modèle linéaire privé de l'observation i. Ces résidus ont une même variance égale à l'unité. Ils se construisent simplement en deux étapes :

1. nous estimons les paramètres β et σ^2 avec tous les individus, excepté le i^{e}, nous obtenons alors $\hat{\beta}_{(i)}$ et $\hat{\sigma}^2_{(i)}$;

2. nous prévoyons y_i par $\hat{y}_i^p = x_i'\hat{\beta}_{(i)}$.

Ce procédé peut paraître long mais dans le cas de la régression linéaire, il est possible de calculer le numérateur avec toutes les données en utilisant la formule liant l'erreur de prévision à l'erreur d'ajustement (ou résidu)

$$y_i - \hat{y}_i^p = \frac{y_i - \hat{y}_i}{1 - h_{ii}}. \tag{3.2}$$

Sous l'hypothèse de normalité des résidus, nous avons le théorème suivant :

Théorème 3.1 (Loi des résidus studentisés par VC)
Si la matrice X est de plein rang, si les $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ et si la suppression de la ligne i ne modifie pas le rang de la matrice, alors les résidus studentisés par VC, notés t_i^*, suivent une loi de Student à $(n-p-1)$ degrés de liberté.

L'exercice 3.4 est dédié à l'analyse des résidus studentisés. Ainsi, les résidus studentisés par VC sont de même variance et peuvent suivre une loi de Student (voir théorème 3.1). Il est donc possible de les analyser simplement. En moyenne, il y a 5 % des observations en dehors de la « bande » de largeur constante qu'une règle « empirique » habituelle fixe à ± 2, car 2 est proche du quantile à 97.5 % d'une loi normale.

Remarque
Nous préconisons d'analyser les résidus dont la valeur est plus grande que le fractile d'ordre $1/n$. Ainsi si l'échantillon vaut 100, nous nous intéresserons aux résidus dont la valeur absolue est supérieure à 2.3, si n vaut 1000, aux résidus dont la valeur absolue est supérieure à 3... En effet il y aura toujours en moyenne des résidus à l'extérieur de l'intervalle $[-2, 2]$.

Conclusion
Les résidus utilisés sont en général les $\hat{\varepsilon}_i$ mais leur variance dépend de l'observation i *via* la matrice de projection. L'utilisation de ces résidus est, à notre avis, à déconseiller. Nous préférons travailler avec des résidus homoscédastiques et donc utiliser t_i ou t_i^*. Ces derniers permettent de détecter des valeurs aberrantes. Il semble cependant préférable d'utiliser t_i^* pour plusieurs raisons :

- les t_i^* suivent un $\mathcal{T}_{n-p-1}$, ils permettent de mieux appréhender une éventuelle non-indépendance non prise en compte par le modèle ;
- nous avons $t_i^* = t_i\sqrt{(n-p-1)/(n-p-t_i^2)}$ et donc lorsque t_i est supérieur à 1, $t_i^* > t_i$ car $\sqrt{(n-p-1)/(n-p-t_i^2)} > 1$. Les résidus studentisés font mieux ressortir les grandes valeurs et permettent donc une détection plus facile des valeurs aberrantes ;
- enfin $\hat{\sigma}_{(i)}$ est indépendant de y_i et n'est donc pas influencé par des erreurs grossières sur la i^{e} observation.

3.1.2 Ajustement individuel au modèle, valeur aberrante

Pour analyser la qualité de l'ajustement d'une observation, il suffit de regarder le résidu correspondant à cette observation. Si ce résidu est anormalement élevé (sens

que nous allons préciser), alors l'individu i est appelé individu aberrant ou point aberrant. Il convient alors d'essayer d'en comprendre la raison (erreur de mesure, individu provenant d'une sous-population) et éventuellement d'éliminer ce point car il peut modifier les estimations.
Une valeur aberrante est une observation qui est mal expliquée par le modèle et admet un résidu élevé. Cette notion est définie par :

Définition 3.1 (Valeur aberrante)
Une donnée aberrante est un point (x'_i, y_i) *pour lequel la valeur associée à* t_i^* *est élevée (comparée au seuil donné par la loi du Student) :* $|t_i^*| > t_{n-p-1}(1-\alpha/2)$.

Il faut faire preuve de bon sens comme indiqué dans la remarque précédente. Généralement les données aberrantes sont détectées en traçant les t_i^*. La détection des données aberrantes ne dépend que de la grandeur des résidus. Voyons cela sur un exemple simulé.

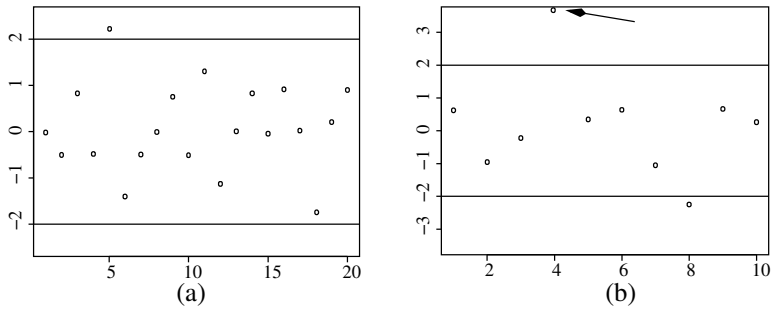


Fig. 3.1 – Résidus studentisés corrects (a) et résidus studentisés avec un individu aberrant à vérifier signalé par une flèche (b) et un second moins important.

La figure (3.1.a) montre un ajustement individuel satisfaisant. Remarquons qu'en théorie $\alpha\%$ des observations sont des valeurs aberrantes. Nous cherchons donc plutôt les résidus dont les valeurs absolues sont nettement au-dessus du seuil de $t_{n-p-1}(1-\alpha/2)$. Ainsi nous nous intéresserons dans la figure (3.1.b) au seul individu désigné par une flèche.
Une fois repérées et notées, il est bon de comprendre pourquoi ces valeurs sont aberrantes : est-ce une erreur de mesure ou d'enregistrement ? Proviennent-elles d'une autre population ?... Nous recommandons d'enlever ces points de l'analyse. Si vous souhaitez les conserver malgré tout, il est indispensable de s'assurer que ce ne sont pas des valeurs influentes : les coefficients et les interprétations tirées du modèle ne doivent pas trop varier avec ou sans ces observations.

3.1.3 Analyse de la normalité

L'hypothèse de normalité sera examinée à l'aide d'un graphique comparant les quantiles des résidus estimés à ces mêmes quantiles sous l'hypothèse de normalité.

Ce type de graphique est appelé Q-Q plot. Supposons que nous ayons n observations $\varepsilon_1, \ldots, \varepsilon_n$ de la variable aléatoire ε qui suit une loi normale $\mathcal{N}(0,1)$. Classons les ε_i par ordre croissant, $\varepsilon_{(1)}, \ldots, \varepsilon_{(n)}$. L'espérance de $\varepsilon_{(i)}$ est alors approchée par

$$\begin{array}{ll} \Phi^{-1}\left(\dfrac{i-3/8}{n+1/4}\right) & \text{si } n \leq 10 \\ \Phi^{-1}\left(\dfrac{i-1/2}{n}\right) & \text{sinon,} \end{array}$$

où $\Phi(.)$ est la fonction de répartition de la loi normale (bijection de $\mathbb{R}$ dans $]0;1[$). Le graphique est alors obtenu en dessinant $\varepsilon_{(1)}, \ldots, \varepsilon_{(n)}$ contre leur espérance théorique respective sous hypothèse de normalité. Si cette hypothèse est respectée, le graphique obtenu sera proche de la première bissectrice (voir fig. 3.10).

3.1.4 Analyse de l'homoscédasticité

Il n'existe pas de procédure précise pour vérifier l'hypothèse d'homoscédasticité. Nous proposons plusieurs graphiques possibles pour détecter une hétéroscédasticité. Il est recommandé de tracer les résidus studentisés par validation croisée t_i^* en fonction des valeurs ajustées $\hat{y}_i$, c'est-à-dire tracer les couples de points $(\hat{y}_i, t_i^*)$. Si une structure apparaît (tendance, cône, vagues), l'hypothèse d'homoscédasticité risque fort de ne pas être vérifiée. Voyons cela sur un graphique.

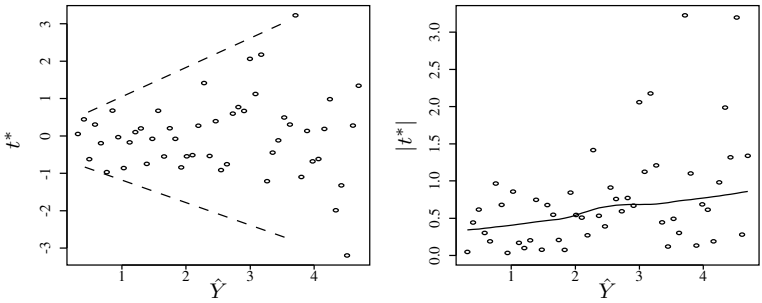


Fig. 3.2 – Hétéroscédasticité des résidus.

L'ajustement n'est pas satisfaisant (fig. 3.2) car la variabilité des résidus augmente avec la valeur de $\hat{y}_i$, on parle de cône de variance croissante avec la valeur de l'axe des abscisses $\hat{Y}$. Le second graphique trace la valeur absolue du résidu avec une estimation de la tendance des résidus. Cette estimation de la tendance est obtenue par un lisseur, ici `lowess` (Cleveland, 1979). Ce lisseur, qui est aussi nommé `loess`, est le plus utilisé pour obtenir ce type de courbe. Il consiste en une régression par polynômes locaux itérée. Nous voyons que la tendance est croissante et donc que la variance des résidus augmente le long de l'axe des abscisses. Ce deuxième graphique permet de repérer plus facilement que le premier les changements de variance éventuels dans les résidus. Le choix de l'axe des abscisses est très important et permet (ou non) de détecter une hétéroscédasticité. D'autres

choix que $\hat{Y}$ en abscisse peuvent s'avérer plus pertinents selon le problème comme le temps, l'indice...
Remédier à l'hétéroscédasticité est compliqué car elle peut avoir plusieurs sources potentielles : variable manquante, effet temporel, effet spatial, variable potentiellement à transformer grâce à l'analyse des résidus partiels (voir section 3.4.3).

3.1.5 Analyse de la structure des résidus

Les résidus sont supposés être non corrélés entre eux ($\mathcal{H}_2$) ou indépendants ($\mathcal{H}_3$). Il existe de nombreuses raisons qui font que les résidus sont corrélés : mauvaise modélisation, structuration temporelle, structuration spatiale... que nous allons analyser *via* des représentations graphiques adaptées.

Structure due à une mauvaise modélisation

Une structure dans les résidus peut être due à une mauvaise modélisation. Supposons que nous ayons oublié une variable intervenant dans l'explication de la variable Y. Cet oubli se retrouvera forcément dans les résidus qui sont par définition les observations moins l'estimation par le modèle. L'hypothèse d'absence de structuration ($\mathrm{Cov}(\varepsilon_i, \varepsilon_j) = 0 \ \forall i \neq j$) risque de ne pas être vérifiée. En effet, la composante oubliée dans le modèle va s'additionner au vrai bruit et devrait apparaître dans le dessin des résidus. Une forme quelconque de structuration dans les graphiques des résidus sera annonciatrice d'un mauvais ajustement du modèle. La figure (3.3) montre les graphiques d'un modèle linéaire $Y = \alpha + \beta_1 X_1 + \varepsilon$ alors que le vrai modèle est un modèle à deux variables $Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$.

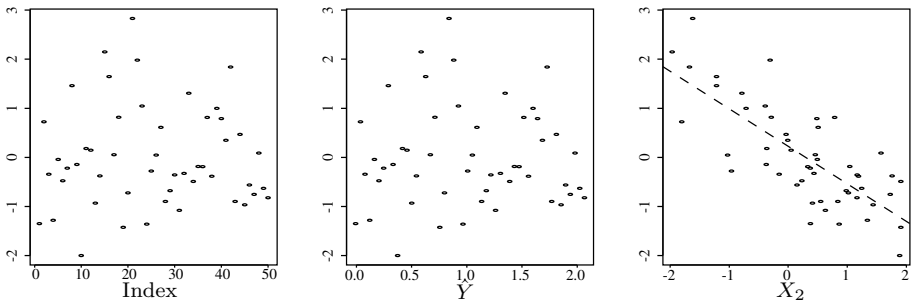


Fig. 3.3 – Les résidus studentisés (par VC) sont représentés comme fonctions du numéro de l'observation (index), de l'estimation du modèle $\hat{Y}$ et de X_2.

L'ajustement paraît non satisfaisant puisqu'une structure linéaire décroissante se dégage des résidus de la troisième représentation. Notons l'importance du choix de l'axe des abscisses car les premiers graphiques, représentant les mêmes résidus, ne laissent pas soupçonner cette tendance décroissante. Le modèle linéaire proposé n'est pas judicieux, il serait bon dans cet exemple simple de rajouter au modèle la variable « oubliée » X_2.

Ce type de diagnostic peut être insuffisant. Une autre méthode plus précise consiste à analyser, variable par variable, si la variable explicative considérée agit bien de manière linéaire sur la variable à expliquer comme cela est requis dans le modèle. Ce type d'analyse sera mené avec des résidus appelés résidus partiels (ou résidus partiels augmentés) ou encore par des régressions partielles. Ces graphiques permettent de constater si une variable candidate est bien utile au modèle et de trouver d'éventuelles fonctions non linéaires de variables explicatives déjà présentes. Rappelons qu'une fonction non linéaire f fixée d'une variable explicative X_j est considérée comme une variable explicative à part entière $X_{p+1} = f(X_j)$ (voir p. 34). Nous verrons cela à la fin de ce chapitre (voir section 3.4.3).

Structure temporelle

Si l'on soupçonne une structuration temporelle (autocorrélation des résidus), un graphique temps en abscisse, résidus en ordonnée sera indiqué. Le test généralement utilisé est le test de Durbin-Watson. Il consiste à tester H_0 : l'indépendance, contre H_1 : les résidus sont non indépendants et suivent un processus autorégressif d'ordre 1 (Montgomery *et al.*, 2001). Il existe cependant de nombreux autres modèles de non-indépendance qui ne seront pas forcément détectés par ce test.
L'utilisation d'un lisseur peut permettre de dégager une éventuelle structuration dans les résidus (voir fig. 3.4) de façon simple et rapide.
Il est cependant difficile, voire impossible, de discerner entre une structuration due à un oubli dans la modélisation de la moyenne et une structuration due à une mauvaise modélisation de la variance comme cela est illustré sur la figure 3.4).

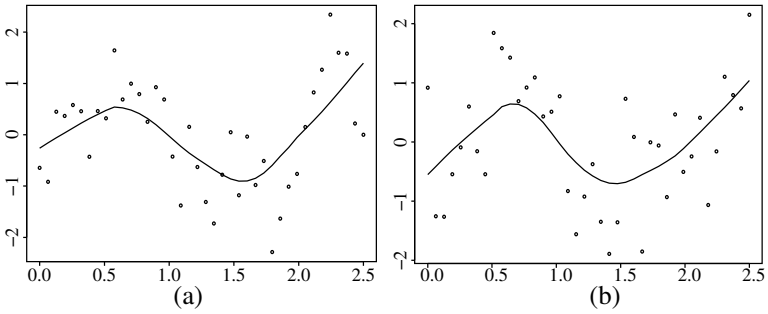


Fig. 3.4 – Graphique (a) variance mal modélisée (tendance sinusoïdale due à un bruit autorégressif d'ordre 1 : $\varepsilon_i = \rho\varepsilon_{i-1} + \eta_i$). Graphique (b) moyenne mal modélisée (composante explicative non prise en compte : $X_2 = 0.2\sin(3x)$).

Structure spatiale

Lorsque les données possèdent une structure spatiale (coordonnées GPS, villes...), un graphique possible consiste à représenter les résidus sur la carte par un cercle ou un carré (selon le signe du résidu estimé) de taille variable (selon la valeur absolue du résidu estimé). Ce type de graphique permettra de détecter une éventuelle

structuration spatiale (agrégats de ronds ou de carrés, ou au contraire alternance des ronds/carrés). Si une structuration est observée, un travail sur les résidus et en particulier sur leur covariance est nécessaire.
Un exemple très classique de structuration est tiré d'Upton & Fingleton (1985). Le but de la modélisation est d'expliquer le nombre de plantes endémiques observées Y par la surface de l'unité de mesure, l'altitude et la latitude. Les résidus studentisés sont représentés sur la carte géographique des emplacements de mesure (fig. 3.5).

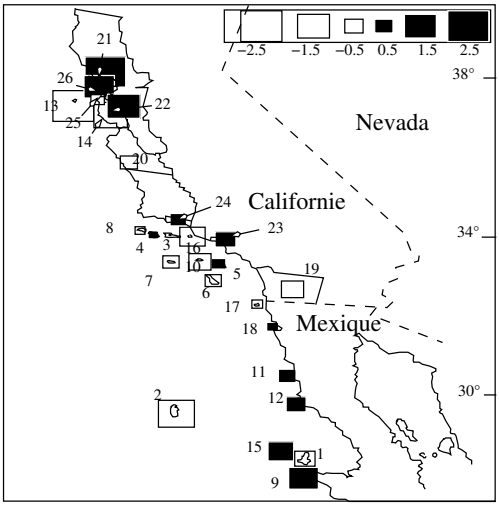


Fig. 3.5 – Exemple de résidus studentisés structurés spatialement.

On observe des agrégats de résidus positifs ou négatifs semblant indiquer une structuration spatiale dans les résidus. Dans cet exemple, une représentation des résidus en fonction de $\hat{Y}$ ou du numéro de l'observation apporte peu d'information. Le choix d'une représentation adéquate est crucial pour faire les bons diagnostics.

Conclusion

Il est impératif de tracer un graphique avec en ordonnée les résidus et en abscisse soit $\hat{Y}$, soit le numéro de l'observation, soit le temps ou tout autre facteur potentiel de non indépendance. Ce type de graphique permettra : de vérifier l'ajustement global, de repérer les points aberrants, ainsi que de vérifier les hypothèses concernant la structure de variance du vecteur ε. D'autres graphiques, comme ceux présentant la valeur absolue des résidus en ordonnée permettront de regarder la structuration de la variance. L'analyse des résidus permet de détecter des différences significatives entre les valeurs observées et les valeurs prédites. Cela permet donc de connaître les points mal prédits et les faiblesses du modèle en termes de moyenne ou de variance. Cependant, cela ne nous renseigne pas sur les variations des estimateurs des paramètres dues à la suppression d'une observation et donc à la robustesse de ces estimations. Les mesures permettant d'étudier cette robustesse sont présentées dans la prochaine section.

3.2 Analyse de la matrice de projection

La matrice de projection

$$P_X = X(X'X)^{-1}X',$$

est la matrice intervenant dans le calcul des valeurs ajustées. En effet,

$$\hat{Y} = P_X Y.$$

Pour la ligne i, en notant h_{ij} l'élément courant de la matrice de projection P_X, cela s'écrit

$$\hat{y}_i = \sum_{j=1}^{n} h_{ij} y_j = h_{ii} y_i + \sum_{j \neq i} h_{ij} y_j.$$

Cette dernière écriture permet de mesurer le poids de l'observation sur son propre ajustement *via* h_{ii}.

Définition 3.2 (Poids de l'observation i)
Le « poids » de l'observation i sur sa propre estimation vaut h_{ii}.

La matrice de la projection orthogonale P_X sur l'espace engendré par les colonnes de X, d'élément courant h_{ij}, admet en particulier comme propriétés (voir exercice 3.2) que si $h_{ii} = 1$ alors $h_{ij} = 0$ pour tout j différent de i et si $h_{ii} = 0$, alors $h_{ij} = 0$ pour tout j différent de i. Nous avons alors les cas extrêmes suivants :
— si $h_{ii} = 1$, $\hat{y}_i$ est entièrement déterminée par y_i car $h_{ij} = 0$ pour tout j ;
— si $h_{ii} = 0$, y_i n'a pas d'influence sur $\hat{y}_i$ (qui vaut alors zéro).
Nous savons aussi que $\mathrm{tr}(P_X) = \sum h_{ii} = p$, la moyenne des h_{ii} vaut donc p/n. Ainsi si h_{ii} est « grand », y_i influe fortement sur $\hat{y}_i$. Différents auteurs ont travaillé sur ce critère et la définition suivante rapporte leur définition de « grand ».

Définition 3.3 (Point levier)
Un point i est un point levier si la valeur h_{ii} de la matrice de projection dépasse les valeurs suivantes :
— $h_{ii} > 2p/n$ *selon Hoaglin & Welsch (1978) ;*
— $h_{ii} > 3p/n$ *pour* $p > 6$ *et* $n - p > 12$ *selon Velleman & Welsh (1981) ;*
— $h_{ii} > 0.5$ *selon Huber (1981).*

Pour un modèle de régression simple dont le nuage de points est représenté sur la figure (3.6) le point désigné par une flèche est un point levier. Sa localisation sur l'axe x est différente des autres points et son poids h_{ii} est prépondérant et supérieur aux valeurs seuils de $2p/n$ et $3p/n$. Cette notion de levier h_{ii} correspond à l'éloignement du centre de gravité de la i^{e} ligne de X. Plus le point est éloigné, plus la valeur des h_{ii} augmente. Remarquons que ce point est levier mais non aberrant car il se situe dans le prolongement de la droite de régression et donc son résidu sera faible. Cependant dans ce cas précis, ce point levier affectera peu l'estimation du paramètre mais il interviendra dans l'estimation de la variance.

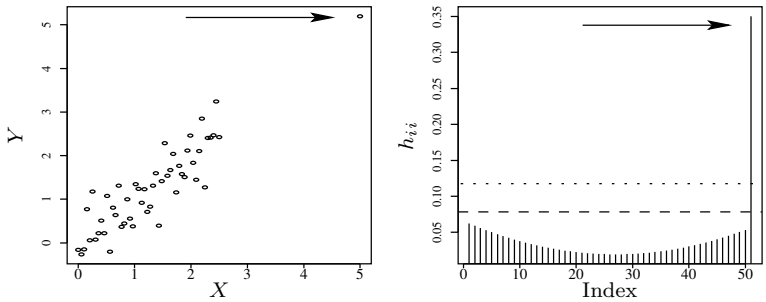


Fig. 3.6 – Exemple d'un point levier, figuré par la flèche, pour un modèle de régression simple. Quantification par h_{ii} de la notion de levier. La ligne en pointillé représente le seuil de $3p/n$ et celle en tiret le seuil de $2p/n$.

Les points leviers sont donc des points atypiques au niveau des variables explicatives. Là encore il est bon de les repérer et de les noter, puis de comprendre pourquoi ces points sont différents : erreur de mesure, erreur d'enregistrement, ou appartenance à une autre population. Il faut aussi se poser la question de la validité du modèle jusqu'à ces points extrêmes. Peut-être aurait-on, avec plus de mesures autour de ces points, un modèle qui changerait, annonçant un modèle différent pour cette population ? Après mûre réflexion ces valeurs pourront être éliminées ou conservées. Dans le premier cas, aucun risque n'est pris au bord du domaine, quitte à sacrifier quelques points. Dans le second cas, le modèle est étendu de manière implicite jusqu'à ces points. Dans certains problèmes, en analysant les points leviers, on peut retrouver deux (ou plusieurs) groupes de valeurs et donc deux (ou plusieurs) sous-populations, il est alors important de se poser la question de la pertinence d'un seul modèle pour toutes les données ou de faire plusieurs sous-modèles.
L'analyse des résidus permet de trouver des valeurs atypiques en fonction de la valeur de la variable à expliquer. L'analyse de la matrice de projection permet de trouver des individus atypiques en fonction des valeurs des variables explicatives (observations éloignées de la moyenne). D'autres critères vont combiner ces deux analyses.

3.3 Autres mesures diagnostiques

La distance de Cook mesure l'influence de l'observation i sur l'estimation du paramètre β. Pour bâtir une telle mesure, nous considérons la distance entre l'estimateur des moindres carrés $\hat{\beta}$ et l'estimateur des moindres carrés $\hat{\beta}_{(i)}$ calculé sans la i^{e} observation. Si la distance est grande, alors l'observation i influence beaucoup l'estimation de β, puisque sa présence ou son absence conduit à des estimations éloignées. $\hat{\beta}$ et $\hat{\beta}_{(i)}$ étant dans $\mathbb{R}^p$, une distance bâtie sur un produit scalaire s'écrit

$$d(\hat{\beta}_{(i)}, \hat{\beta}) \quad = \quad (\hat{\beta}_{(i)} - \hat{\beta})' Q (\hat{\beta}_{(i)} - \hat{\beta}),$$

où Q est une matrice symétrique définie positive. L'équation donnant une région de confiance simultanée (voir 5.4, p. 94)

$$\mathrm{RC}_\alpha(\beta) \quad = \quad \left\{\beta \in \mathbb{R}^p, \quad \frac{1}{p\hat{\sigma}^2}(\hat{\beta}-\beta)'(X'X)(\hat{\beta}-\beta) \leq f_{p,n-p}(1-\alpha)\right\},$$

permet de dire que dans 95 % des cas, la distance (associée à $Q = X'X/p\hat{\sigma}^2$) entre β et $\hat{\beta}$ est inférieure à $f_{p,n-p}(1-\alpha)$. Par analogie, nous pouvons donc utiliser cette distance, appelée distance de Cook, pour mesurer l'influence de l'observation i dans le modèle.

Définition 3.4 (Distance de Cook)
La distance de Cook pour la ième observation est donnée par

$$C_i \quad = \quad \frac{1}{p\hat{\sigma}^2}(\hat{\beta}_{(i)}-\hat{\beta})'(X'X)(\hat{\beta}_{(i)}-\hat{\beta}).$$

Elle peut se récrire de manière plus concise et plus simple à calculer :

$$C_i = \frac{1}{p}\frac{h_{ii}}{1-h_{ii}}t_i^2 = \frac{h_{ii}}{p(1-h_{ii})^2}\frac{\hat{\varepsilon}_i^2}{\hat{\sigma}^2}.$$

Une observation influente est donc une observation qui, enlevée, conduit à une grande variation dans l'estimation des coefficients, c'est-à-dire à une distance de Cook élevée. Pour juger si la distance C_i est élevée, Cook (1977) propose le seuil $f_{p,n-p}(0.1)$ comme souhaitable et le seuil $f_{p,n-p}(0.5)$ comme préoccupant. Certains auteurs citent comme seuil la valeur 1, approximation raisonnable de $f_{p,n-p}(0.5)$. La distance de Cook (deuxième définition) peut être vue comme la contribution de deux termes. Le carré du résidu standardisé t_i^2 (voir (3.1)) mesure le degré d'adéquation de l'observation y_i au modèle estimé $x_i'\hat{\beta}$. Le second terme correspond au rapport $\mathrm{V}(\hat{y}_i)/\mathrm{V}(\hat{\varepsilon}_i)$ et mesure la sensibilité de l'estimateur $\hat{\beta}$ à l'observation i. La distance de Cook mesure deux caractères en même temps : le caractère aberrant quand t_i est élevé et le caractère levier lorsque $\mathrm{V}(\hat{y}_i)/\mathrm{V}(\hat{\varepsilon}_i) = h_{ii}/1-h_{ii}$ est élevé. Les points présentant des distances de Cook élevées seront des points aberrants, ou leviers, ou les deux, et influenceront l'estimation puisque la distance de Cook est une distance entre $\hat{\beta}$ et $\hat{\beta}_{(i)}$.
A l'image des points aberrants et leviers, nous recommandons de supprimer les observations présentant une forte distance de Cook. Si l'on souhaite toutefois absolument garder ces points, il sera très important de vérifier que les coefficients estimés et les interprétations tirées du modèle ne varient pas trop avec ou sans ces observations influentes.
La figure (3.7) représente le nuage de points, les résidus studentisés et la distance de Cook correspondant au modèle de régression simple pour les points de la figure (3.6). Nous voyons que des points admettant de forts résidus (points éloignés de la droite) possèdent une distance de Cook élevée (cas des points 4, 6, 12, 29, 44 et 45). Mais les points leviers possèdent un rapport $h_{ii}/(1-h_{ii})$ élevé, par définition. Le point 51 bien qu'ayant un résidu faible apparaît comme ayant une distance de

Cook relativement forte (la 8e plus grande). Cela illustre bien que la distance de Cook opère un compromis entre points aberrants et points leviers. Notons encore une fois que le point 51 n'est ni influent ni aberrant. Son résidu t^*_{51} n'est pas élevé et il se situe dans le prolongement de l'axe du nuage, ce qui veut dire que la droite ajustée par MC sans ce point est voisine de la droite ajustée par MC avec ce point.

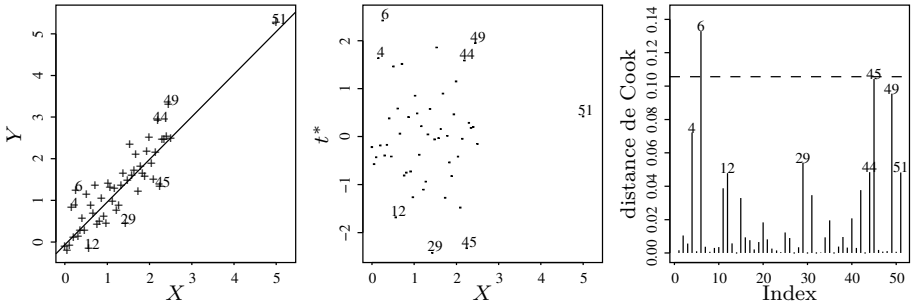


Fig. 3.7 – Exemple du point levier (le 51). Les points associés aux 8 plus grandes valeurs de la distance de Cook sont numérotés ainsi que leur distance de Cook et leurs résidus studentisés (par VC). La droite en trait plein est la droite des MC.

Notons enfin que les seuils de la distance de Cook sont $f_{p,n-p}(0.5) = 0.7$ et $f_{p,n-p}(0.1) = 0.11$, ce dernier figurant en pointillé sur le graphique (3.7). Ici les distances de Cook semblent assez bien réparties au niveau hauteur et aucun point ne se détache nettement.
En utilisant encore les mêmes 50 points mais en remplaçant le point levier par un point clairement aberrant (mais non levier), nous voyons que ce nouveau point 51 est bien aberrant (fig. 3.8), son résidu t^*_{51} est en effet très élevé.

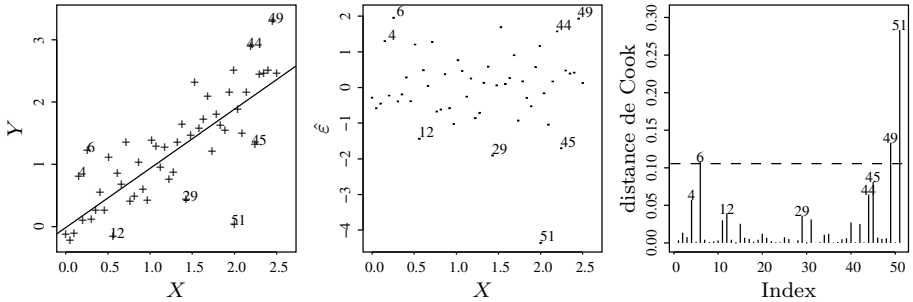


Fig. 3.8 – Exemple du point fortement aberrant (numéro 51). Les points associés aux 8 plus grandes valeurs de la distance de Cook sont numérotés, ainsi que leur distance de Cook et leurs résidus studentisés (par VC). La droite en trait plein est la droite ajustée par MC.

La distance de Cook, malgré la position de ce point 51 vers le milieu des x, est élevée et cela uniquement à cause de son caractère aberrant. Bien entendu un point peut être à la fois levier et aberrant. Ici la distance de Cook du point 51 se détache

nettement, indiquant que ce point pourrait être éventuellement supprimé. Le seuil de $f_{p,n-p}(0.5)$ semble assez conservateur.
Une autre mesure d'influence est donnée par la distance de Welsh-Kuh. Si l'on reprend la définition de la distance de Cook pour l'observation i, elle s'écrit comme $(\hat{y}_i - x_i'\hat{\beta}_{(i)})^2/\hat{\sigma}^2$ à $1/p$ près. Cela représente le carré de l'écart entre $\hat{y}_i$ et sa prévision $\hat{y}_i^p$ divisé par la variance estimée de l'erreur. Il faut donc utiliser un estimateur de σ^2. Si l'on utilise l'estimateur classique $\hat{\sigma}^2$, alors une observation influente risque de « perturber » l'estimation $\hat{\sigma}^2$. Il est donc préférable d'utiliser $\hat{\sigma}^2_{(i)}$.

Définition 3.5 (DFFITS)
L'écart de Welsh-Kuh, souvent appelé DFFITS dans les logiciels, est défini par

$$Wk_i = |t_i^*|\sqrt{\frac{h_{ii}}{1-h_{ii}}}.$$

Cette quantité permet d'évaluer l'écart standardisé entre l'estimation bâtie sur toutes les observations et l'estimation bâtie sur toutes les observations sauf la i^{e}. Cet écart de Welsh-Kuh mesure ainsi l'influence simultanée d'une observation sur l'estimation des paramètres β et σ^2. Si l'écart de Welsh-Kuh est supérieur à $2\sqrt{p+1}/\sqrt{n}$ en valeur absolue, alors il est conseillé d'analyser les observations correspondantes.
D'autres mesures diagnostiques sont données dans le livre d'Antoniadis *et al.* (1992, pp 36-40). En guise de remarque finale, la régression robuste est une alternative très intéressante si le problème des observations influentes s'avère délicat (Rousseeuw & Leroy, 1987).

3.4 Effet d'une variable explicative

3.4.1 Ajustement au modèle

Nous désirons savoir si la modélisation de l'espérance de Y par $X\beta$, estimée par $X\hat{\beta}$, est correcte. Le modèle est-il satisfaisant ou ne faudrait-il pas rajouter de nouvelles variables explicatives ou de nouvelles fonctions fixées des variables explicatives et lesquelles ? Dans ce paragraphe, nous nous posons la question de la qualité d'ajustement du modèle pour une variable explicative X_j donnée, ce qui revient aux trois questions suivantes :

1. cette variable X_j est-elle utile ?

2. est-ce que cette variable agit linéairement sur la prévision de Y ou faut-il introduire une transformation de cette variable $f(X_j)$?

3. quelle transformation $f(X_j)$ est à introduire afin d'améliorer le modèle ?

Pour répondre à ces questions, remarquons que l'on peut toujours utiliser les procédures de choix de variables (voir chapitre suivant) ainsi que les tests entre modèles emboîtés :

- si l'on se pose la question de l'utilité de la variable X_j on peut tester

$$\begin{array}{llll} \mathrm{H}_0 : & \beta_j = 0 \quad \text{contre} \quad \mathrm{H}_1 : & \beta_j \neq 0 \quad \text{ou écrit différemment} \\ \mathrm{H}_0 : & \mathbb{E}(Y) = \displaystyle\sum_{k=1, k\neq j}^{p} \beta_k X_k \quad \text{contre} \quad \mathrm{H}_1 : & \mathbb{E}(Y) = X\beta. \end{array}$$

- si l'on se pose la question d'une transformation $f(X_j)$ notée X_{p+1} on peut tester

$$\mathrm{H}_0 : \quad \mathbb{E}(Y) = X\beta \quad \text{contre} \quad \mathrm{H}_1 : \quad \mathbb{E}(Y) = X\beta + \beta_{p+1} X_{p+1}.$$

Cependant, sans connaître *a priori* $f(.)$, il est impossible d'effectuer le test. Ce paragraphe va proposer des outils graphiques permettant de répondre à ces trois questions rapidement, en conservant à l'esprit que la première question peut être résolue avec un test que nous verrons plus en détails au chapitre 5.

3.4.2 Régression partielle : impact d'une variable

Afin de connaître l'impact de la j^{e} variable X_j lors d'une régression :

1. nous effectuons d'abord une régression avec les $p-1$ autres variables. Les résidus obtenus correspondent alors à la part de Y qui n'a pas été expliquée par les $p-1$ variables ;

2. la seconde partie de l'analyse correspond alors à l'explication de ces résidus non pas par la variable X_j mais par la part de la variable X_j qui n'est pas déjà expliquée par les $p-1$ autres variables.

Tout d'abord supposons que le modèle complet soit vrai, c'est-à-dire que

$$Y = X\beta + \varepsilon.$$

Afin d'analyser l'effet de la j^{e} variable X_j, partitionnons la matrice X en deux, une partie sans la j^{e} variable que nous notons $X_{\bar{j}}$ et l'autre avec la j^{e} variable X_j.

Le modèle s'écrit alors

$$Y = X_{\bar{j}}\beta_{\bar{j}} + \beta_j X_j + \varepsilon,$$

où $\beta_{\bar{j}}$ désigne le vecteur β privé de sa j^{e} coordonnée notée β_j. Afin de quantifier l'apport de la variable X_j, projetons Y sur l'orthogonal de $\Im(X_{\bar{j}})$. Nous noterons cette projection $P_{X_{\bar{j}}^{\perp}}$. Nous avons alors

$$\begin{aligned} P_{X_{\bar{j}}^{\perp}} Y &= P_{X_{\bar{j}}^{\perp}} X_{\bar{j}}\beta_{\bar{j}} + P_{X_{\bar{j}}^{\perp}} \beta_j X_j + P_{X_{\bar{j}}^{\perp}} \varepsilon \\ P_{X_{\bar{j}}^{\perp}} Y &= \beta_j P_{X_{\bar{j}}^{\perp}} X_j + P_{X_{\bar{j}}^{\perp}} \varepsilon \\ P_{X_{\bar{j}}^{\perp}} Y &= \beta_j P_{X_{\bar{j}}^{\perp}} X_j + \eta. \end{aligned} \tag{3.3}$$

Nous avons donc un modèle de régression dans lequel nous cherchons à expliquer une variable (aléatoire) $P_{X_{\bar{j}}^{\perp}}Y$ par un modèle linéaire dépendant d'une variable fixe $P_{X_{\bar{j}}^{\perp}}X_j$ additionnée à un bruit aléatoire $\eta = P_{X_{\bar{j}}^{\perp}}\varepsilon$.
Cette équation suggère que si le modèle complet est vrai, alors un modèle de régression univariée est valide entre $P_{X_{\bar{j}}^{\perp}}Y$ et $P_{X_{\bar{j}}^{\perp}}X_j$ et donc qu'il suffit de dessiner $P_{X_{\bar{j}}^{\perp}}Y$ en fonction de $P_{X_{\bar{j}}^{\perp}}X_j$ pour le vérifier graphiquement. Ce graphique est appelé graphique de la régression partielle pour la variable X_j :

1. si les points forment une droite de pente $\beta_j \neq 0$, alors le modèle pour la variable X_j est bien linéaire ;

2. si les points forment une droite de pente presque nulle, alors la variable X_j n'a aucune utilité dans le modèle ;

3. si les points forment une courbe non linéaire f, il sera alors utile de remplacer X_j par une fonction non linéaire dans le modèle complet.

Remarquons l'utilité de l'abscisse, qui est $P_{X_{\bar{j}}^{\perp}}X_j$ et non pas directement X_j. Cette abscisse représente la projection de la variable X_j sur les autres variables explicatives $X_{\bar{j}}$, c'est-à-dire la partie de X_j non déjà expliquée linéairement par les autres variables, ou autrement dit la partie de l'information apportée par X_j non déjà prise en compte par le modèle linéaire sans cette variable. Cela permet donc de faire apparaître uniquement la partie non redondante de l'information apportée par X_j pour l'explication linéaire de Y (voir exercice 3.6).

Proposition 3.1 (Régression partielle)
Notons $\tilde{\beta}_j$ l'estimateur des moindres carrés de β_j dans le modèle de régression simple (3.3). Notons $\hat{\beta}_j$ la j^e composante de $\hat{\beta}$, l'estimateur des moindres carrés obtenu dans le modèle complet. Nous avons alors

$$\tilde{\beta}_j = \hat{\beta}_j.$$

3.4.3 Résidus partiels et résidus partiels augmentés

Le problème de l'utilisation du graphique précédent correspond au calcul des abscisses $P_{X_{\bar{j}}^{\perp}}X_j$. Afin de contourner ce problème et d'obtenir un graphique facile à effectuer, nous définissons les résidus partiels :

Définition 3.6 (Résidus partiels)
Les résidus partiels pour la variable X_j sont définis par

$$\hat{\varepsilon}_P^j = \hat{\varepsilon} + \hat{\beta}_j X_j. \tag{3.4}$$

Le vecteur $\hat{\varepsilon}$ correspond aux résidus obtenus avec toutes les variables et $\hat{\beta}_j$ est la j^e coordonnée de $\hat{\beta}$ estimateur des MC obtenu dans le modèle complet.

Un graphique représentant X_j en abscisse et ces résidus partiels en ordonnée aura, si le modèle complet est valide, une allure de droite de pente estimée $\hat{\beta}_j$ par MC.

En effet, la pente de régression univariée estimée par MC est (voir équation 1.4)

$$\frac{\langle \hat{\varepsilon}_P^j, X_j \rangle}{\langle X_j, X_j \rangle} = \frac{\langle \hat{\varepsilon}, X_j \rangle + \hat{\beta}_j \langle X_j, X_j \rangle}{\langle X_j, X_j \rangle} = \frac{\langle P_{X^\perp} Y, X_j \rangle + \hat{\beta}_j \langle X_j, X_j \rangle}{\langle X_j, X_j \rangle} = \hat{\beta}_j.$$

Il est en général préférable d'enlever l'information apportée par la moyenne commune à chacune des variables et de considérer ainsi les variables centrées et les résidus partiels correspondants

$$\hat{\varepsilon}_P^j = \hat{\varepsilon} + \bar{y}\mathbb{1} + \hat{\beta}_j (X_j - \bar{X}_j),$$

où $\bar{X}_j$ est le vecteur de $\mathbb{R}^n$ ayant toujours la même coordonnée : $\sum_{i=1}^n x_{ij}/n$.
Les graphiques des résidus partiels sont à l'image de ceux des régressions partielles, ils comportent pour chaque variable X_j en abscisse cette variable X_j et en ordonnée les résidus partiels correspondants $\hat{\varepsilon}_P^j$. Si le modèle complet est vrai, le graphique montre une tendance linéaire et la variable X_j intervient bien de manière linéaire. Si par contre la tendance sur le graphique est non linéaire selon une fonction $f(.)$, il sera bon de remplacer X_j par $f(X_j)$.
Le fait d'utiliser X_j en abscisse pour les graphiques des résidus partiels permet de trouver beaucoup plus facilement la transformation $f(X_j)$ que dans les graphiques des régressions partielles correspondants. Par contre, en n'enlevant pas à X_j l'information déjà apportée par les autres variables, la pente peut apparaître non nulle alors que l'information supplémentaire apportée par X_j par rapport à $X_{\bar{j}}$ n'est pas importante. Cela peut se produire lorsque X_j est très corrélée linéairement à une ou plusieurs variables de $X_{\bar{j}}$. On peut alors avoir recours à une procédure de test ou de sélection de modèle dans ce cas. Si l'objectif est de vérifier que la variable X_j agit linéairement dans le modèle et de vérifier qu'aucune transformation non linéaire $f(X_j)$ n'améliorera le modèle, il est alors préférable d'utiliser les résidus partiels.
Des résultats empiriques ont montré que, dans certaines situations, les résidus partiels augmentés (Mallows, 1986) peuvent être meilleurs que les résidus partiels.

Définition 3.7 (Résidus partiels augmentés)
Les résidus partiels augmentés pour la variable X_j sont définis par

$$\hat{\varepsilon}_{AP}^j = \hat{\varepsilon}^\star + \hat{\alpha}_j X_j + \hat{\alpha}_{p+1} X_j^2,$$

où $\hat{\varepsilon}^\star = \hat{Y}^\star - Y$ et $\hat{Y}^\star = (X|X_j^2)\hat{\alpha}$ est l'estimation de Y par le modèle complet augmenté d'un terme quadratique $Y = X_1\beta_1 + \ldots + X_p\beta_p + X_j^2\beta_{p+1} + \varepsilon$.

On peut encore utiliser une autre version sans l'effet de la moyenne

$$\hat{\varepsilon}_{AP,i}^j = \hat{\varepsilon}^\star + \bar{y} + \hat{\alpha}_j (X_{ij} - \bar{X}_j) + \hat{\alpha}_{p+1} \left[(X_{ij} - \bar{X}_j)^2 - \frac{1}{n} \sum_{i=1}^n (X_{ij} - \bar{X}_j)^2 \right].$$

Nous renvoyons le lecteur intéressé par l'heuristique de ces résidus partiels à l'article de Mallows (1986).

3.5 Exemple : la concentration en ozone

Revenons à l'exemple de la prévision des pics d'ozone. Nous expliquons le pic d'ozone `O3` par 6 variables : la teneur maximum en ozone la veille (`O3v`), la température prévue par Météo France à 6 h (`T6`), à midi (`T12`), une variable synthétique (la projection du vent sur l'axe est-ouest notée `Vx`) et enfin les nébulosités prévues à midi (`Ne12`) et à 15 h (`Ne15`). Nous avons pour ce travail $n = 1014$ observations.

```
> ozone <- read.table("ozone_long.txt", header = T, sep = ";")
> p <- ncol(ozone) ; n <- nrow(ozone)
> mod.lin6v <- lm(O3~T6 + T12 + Ne12 + Ne15 + Vx + O3v,data=ozone)
```

Commençons par représenter les résidus studentisés en fonction du numéro d'observation qui correspond ici à l'ordre chronologique.

```
> plot(rstudent(mod.lin6v), pch = ".",
+            ylab = "Résidus studentisés par VC")
> abline(h = c(-2,2))
> lines(lowess(rstudent(mod.lin6v)))
```

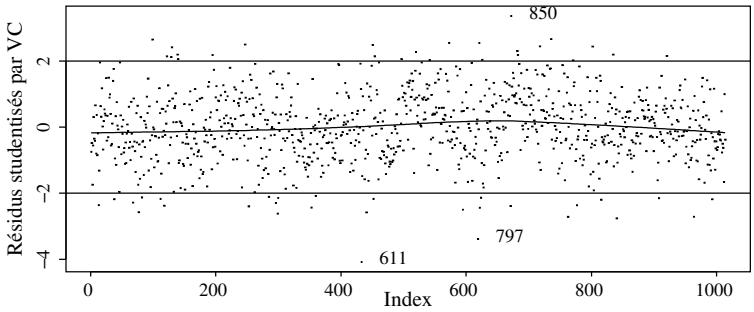


Fig. 3.9 – Résidus studentisés par VC du modèle de régression à 6 variables.

Les résidus studentisés (fig. 3.9) font apparaître une structuration presque négligeable en forme de sinusoïde en fonction du numéro des observations, ou du temps, les observations étant rangées par date de mesure. Cela peut paraître courant puisque nous avons des variables mesurées dans le temps et cette légère variation peut être vue comme une autocorrélation (éventuelle) des résidus.
Nous avons 1014 observations, il est normal qu'un certain nombre de résidus apparaissent en dehors de la bande $(-2, 2)$. Seules les 3 observations franchement éloignées de l'axe horizontal (les numéros 611, 797 et 850) peuvent sembler aberrantes. Ces observations sont donc mal expliquées par le modèle à 6 variables.
Une analyse complémentaire sur ces journées pour mieux comprendre ces individus pourrait être entreprise : sont-ils dus à une erreur de mesure, à une défaillance de l'appareillage, à une journée exceptionnelle ou autre ? Ces points sont mal prédits

mais ne sont pas forcément influents et ne faussent donc pas forcément le modèle. Il n'y a donc pas lieu de les éliminer même si l'on sait qu'ils sont mal expliqués. Bien que nous n'utilisions pas l'hypothèse de normalité ici, nous pouvons tracer à titre d'exemple le graphique Quantile-Quantile.

```
> plot(mod.lin6v, which = 2, sub = "", main = "")
> abline(0,1)
```

Nous observons sur le graphique (3.10) que la normalité semble bien respectée, tous les points étant sur la première bissectrice. Nous apercevons encore les points aberrants numéros 611, 797 et 850.

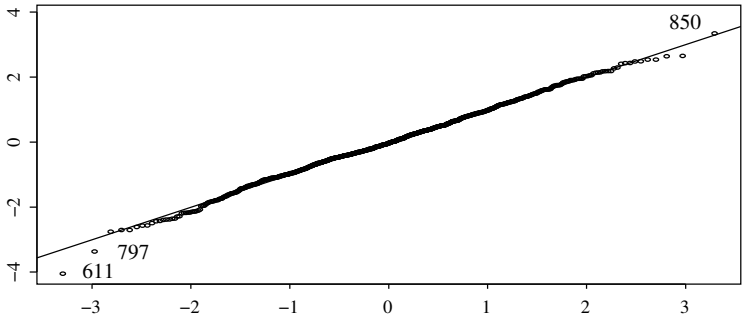
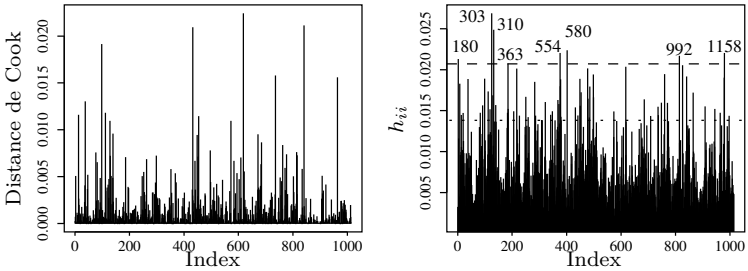


Fig. 3.10 – Q-Q plot pour le modèle à 6 variables explicatives.

Représentons maintenant les points leviers et influents grâce aux codes

```
> plot(cooks.distance(mod.lin6v),type="h",ylab="Distance de Cook")
> seuil1 <- qf(0.1,p,n-p) ; abline(h=seuil1)
> infl.ozone <- influence.measures(mod.lin6v)
> plot(infl.ozone$infmat[,"hat"],type="h",ylab="hii")
> seuil1 <- 3*p/n ; abline(h=seuil1,col=1,lty=2)
> seuil2 <- 2*p/n ; abline(h=seuil2,col=1,lty=3)
```



(a) (b)

Fig. 3.11 – Distance de Cook (a) et points leviers (b).

Aucune observation (fig. 3.11) ne montre une distance de Cook nettement supérieure aux autres et il ne semble pas y avoir d'observation très influente. Le seuil $f_{p,n-p}(0.1) = 0.4$ est supérieur à toutes les observations. Au niveau des points leviers, beaucoup d'individus sont supérieurs au seuil indicatif de $2p/n$, 8 seulement sont au-dessus du seuil de $3p/n$ et enfin aucun n'est aux environs de 0.5. De manière plus générale les h_{ii} sont peu différents les uns des autres, nous conservons toutes les observations. Nous avons vu que le graphique d'ajustement global, résidus studentisés en fonction d'un indice, montre une légère oscillation. Cela peut être dû à une autocorrélation des résidus, donc une mauvaise structure de variance des résidus, qui n'est donc pas diagonale : $\mathrm{V}(\varepsilon) \neq \sigma^2 I_n$. Cependant, cela peut aussi être dû à une mauvaise modélisation de la moyenne. Nous allons donc considérer les graphiques des résidus partiels pour toutes les variables explicatives.
Le graphique des résidus partiels est obtenu grâce aux commandes suivantes (les ordres étant identiques pour chacune des variables, nous ne donnons que ceux concernant la variable `O3v`) :

```
> residpartiels <- resid(mod.lin6v, type = "partial")
> prov <- loess(residpartiels[,"O3v"] ~ ozone$O3v)
> ordre <- order(ozone$O3v)
> plot(ozone$O3v, residpartiels[,"O3v"], pch=".",ylab="",xlab="")
> matlines(ozone$O3v[ordre], predict(prov)[ordre])
> abline(lsfit(ozone$O3v, residpartiels[,"O3v"]), lty = 2)
```

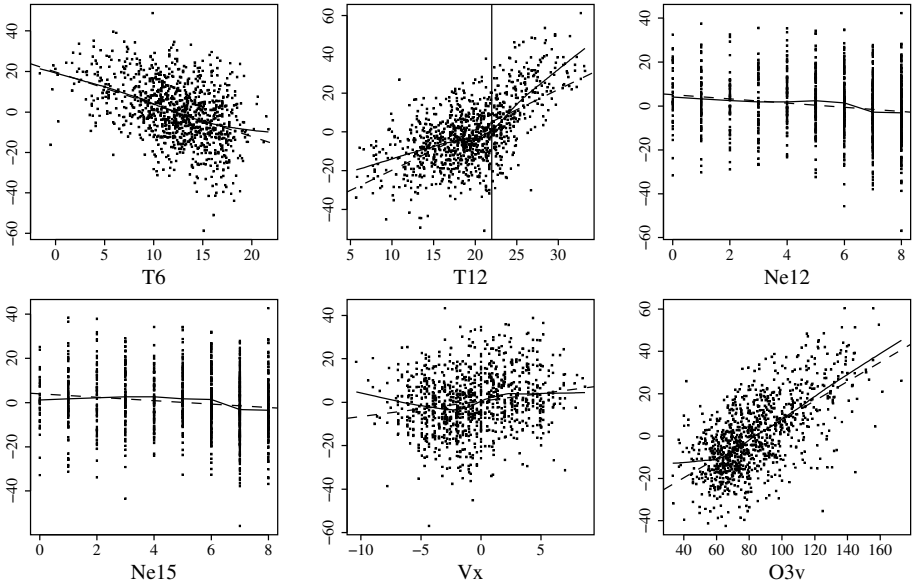


Fig. 3.12 – Résidus partiels pour les 6 variables explicatives. Le trait continu représente le résumé lissé des données par le lisseur `loess`.

Les graphiques des résidus partiels (fig. 3.12) pour les variables `T6`, `Ne12`, `Ne15` et `O3v` montrent qu'aucune transformation n'est nécessaire, les résidus partiels étant répartis le long de la droite ajustée (en pointillé).
Pour la variable `T12` on note que le nuage est réparti en deux sous-ensembles : avant 22 degrés C ou après. Chacun de ces deux sous-ensembles semble être réparti le long d'une droite de pente différente. Nous allons donc ajouter une variable qui va prendre la valeur 0 si `T12`$\leq$ 22 et les valeurs (`T12`-22) si `T12`$>$ 22. Le R^2 passe de 0.669 à 0.708. L'ajustement est donc grandement amélioré par cette variable.
Pour la variable `Vx` nous retrouvons une légère tendance sinusoïdale autour de l'axe des abscisses, indiquant que la variable `Vx` semble avoir peu d'influence. Si l'on ajuste une sinusoïde et que l'on remplace la variable `Vx` par la fonction $f(\texttt{Vx}) = -4.54\cos\{0.45(10.58 - \texttt{Vx})\}$, le R^2 passe à 0.713. Cependant, cette fonction ainsi que la fonction linéaire par morceau pour `T12` *dépendent des données* et ne sont pas des fonctions fixées *a priori* avant le début de l'étude.
Pour toutes les variables, les résidus partiels augmentés offrent exactement les mêmes représentations et ne sont donc pas représentés ici.

3.6 Exercices

Exercice 3.1 (Questions de cours)

1) Lors d'une régression multiple, la somme des résidus vaut zéro :
   A. toujours,
   B. jamais,
   C. cela dépend des variables explicatives utilisées.
2) Les résidus studentisés sont-ils
   A. homoscédastiques,
   B. hétéroscédastiques,
   C. on ne sait pas.
3) Un point levier peut-il être aberrant ?
   A. toujours,
   B. jamais,
   C. parfois.
4) Un point aberrant peut-il être levier ?
   A. toujours,
   B. jamais,
   C. parfois.
5) La distance de Cook est-elle basée sur un produit scalaire ?
   A. oui,
   B. non,
   C. cela dépend des données.

Exercice 3.2 (Propriétés d'une matrice de projection)
Considérons la matrice de projection orthogonale sur l'espace engendré par les colonnes de X P_X de terme courant h_{ij}, montrer que

1) $\mathrm{tr}(P_X) = \sum h_{ii} = p$.
2) $\mathrm{tr}(P_X) = \mathrm{tr}(P_X P_X)$ c'est-à-dire $\sum_i \sum_j h_{ij}^2 = p$.
3) $0 \leq h_{ii} \leq 1$ pour tout i.

4) $-0.5 \leq h_{ij} \leq 0.5$ pour tout j différent de i.
5) Si $h_{ii} = 1$ alors $h_{ij} = 0$ pour tout j différent de i.
6) Si $h_{ii} = 0$, alors $h_{ij} = 0$ pour tout j différent de i.

Exercice 3.3 (Lemme d'inversion matricielle)
Soit M une matrice symétrique inversible de taille $p \times p$, u et v deux vecteurs de taille p. Montrer que

$$\left(M + uv'\right)^{-1} = M^{-1} - \frac{M^{-1}uv'M^{-1}}{1 + u'M^{-1}v}. \tag{3.5}$$

Exercice 3.4 (†Résidus studentisés)
Nous considérons la matrice du plan d'expérience X de taille $n \times p$. Nous notons x_i' la i^{e} ligne de la matrice X et $X_{(i)}$ la matrice X privée de la i^{e} ligne, de taille $(n-1) \times p$.
1) Montrer que $X'X = X_{(i)}'X_{(i)} + x_i x_i'$.
2) Montrer que $X_{(i)}'Y_{(i)} = X'Y - x_i' y_i$.
3) En vous servant de l'équation (3.5), montrer que

$$(X_{(i)}'X_{(i)})^{-1} \quad = \quad (X'X)^{-1} + \frac{1}{1-h_{ii}}(X'X)^{-1}x_i x_i'(X'X)^{-1},$$

où h est le terme courant de la matrice de projection sur $\Im(X)$.
4) Montrer que la prévision de l'observation x_i vaut

$$\hat{y}_i^p \quad = \quad \frac{1}{1-h_{ii}}\hat{y}_i - \frac{h_{ii}}{1-h_{ii}}y_i.$$

5) Montrer que les résidus studentisés par validation croisée définis par :

$$t_i^* = \frac{\hat{\varepsilon}_i}{\hat{\sigma}_{(i)}\sqrt{1-h_{ii}}},$$

où $\hat{\sigma}_{(i)}$ est l'estimateur de σ dans le modèle privé de la i^{e} observation, peuvent s'écrire sous la forme

$$t_i^* = \frac{y_i - \hat{y}_i^p}{\hat{\sigma}_{(i)}\sqrt{1 + x_i'(X_{(i)}'X_{(i)})^{-1}x_i}}.$$

6) Sous l'hypothèse que $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_n)$, quelle est la loi de t_i^* ?

Exercice 3.5 (Distance de Cook)
Nous reprenons les notations et résultats des exercices précédents.
1) Montrer que

$$\hat{\beta}_{(i)} = \hat{\beta} - \frac{1}{1-h_{ii}}(X'X)^{-1}x_i(y_i - x_i'\hat{\beta}).$$

2) Montrer que la distance de Cook définie par

$$C_i = \frac{1}{p\hat{\sigma}^2}(\hat{\beta}_{(i)} - \hat{\beta})'X'X(\hat{\beta}_{(i)} - \hat{\beta}),$$

s'écrit aussi sous la forme

$$C_i = \frac{1}{p}\,\frac{h_{ii}}{1-h_{ii}}\,\frac{\hat{\varepsilon}_i^2}{\hat{\sigma}^2(1-h_{ii})}.$$

Exercice 3.6 (Régression partielle)
Démontrer la proposition (3.1).

Exercice 3.7 (TP : Résidus partiels)
Les données se trouvent dans le fichier tprespartiel.dta et tpbisrespartiel.dta, l'objectif de ce TP est de montrer que l'analyse des résidus partiels peut améliorer la modélisation.

1) Importer les données, vous avez une variable à expliquer Y et quatre variables explicatives.
2) Estimer les paramètres du modèle $Y_i = \beta_0 + \beta_1 X_{i,1} + \cdots + \beta_4 X_{i,4} + \varepsilon_i$.
3) Analyser les résidus partiels `residuals(...,type="partial")`.
4) Que pensez-vous des résultats ?
5) Remplacer X_4 par $X_5 = X_4^2$ dans le modèle précédent. Que pensez-vous de la nouvelle modélisation ? On pourra comparer ce modèle à celui de la question précédente.
6) Analyser les résidus partiels du nouveau modèle.
7) Faire le même travail pour tp2bisrespartiel.

Chapitre 4

Extensions : non-inversibilité et (ou) erreurs corrélées

L'objectif de ce chapitre est de proposer des solutions lorsque les hypothèses classiques du modèle linéaire ne sont pas vérifiées. Ainsi dans les chapitres précédents, nous avons considéré le modèle de régression

$$Y \quad = \quad X\beta + \varepsilon$$

et avons supposé que la matrice X était de plein rang (hypothèse $\mathcal{H}_1$) et que la variance de ε était $\mathrm{V}(\varepsilon) = \sigma^2 I$ (hypothèse $\mathcal{H}_2$). Il existe de nombreux cas où ces hypothèses ne sont pas satisfaites.
Dans une première partie, nous allons traiter le cas où l'hypothèse $\mathcal{H}_1$ n'est pas satisfaite. Dans ce cas la matrice $X'X$ n'est pas inversible et l'estimateur des MCO n'est donc pas calculable. Nous présentons la régression ridge qui permet de pallier à ce problème. Cette méthode sera étudiée dans un contexte plus général au chapitre 8.
Nous traiterons dans une seconde partie le cas où l'hypothèse $\mathcal{H}_2$ n'est pas vérifiée et nous introduirons la méthode des moindres carrés généralisés (MCG).

4.1 Régression ridge

Le problème initial des moindres carrés ordinaires consiste à chercher le vecteur de coefficient $\hat{\beta}$ qui minimise les moindres carrés ordinaires, c'est-à-dire :

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|Y - X\beta\|^2. \tag{4.1}$$

L'hypothèse $\mathcal{H}_1$ (la matrice X est de plein rang) permet alors de trouver une unique solution au problème posé, à savoir $\hat{\beta} = (X'X)^{-1}X'Y$.
Cependant, dans de nombreux cas, cette hypothèse n'est pas satisfaite. Ainsi, par exemple, si $p > n$ il y a plus de coefficients à estimer que d'observations et la

matrice X n'est plus de plein rang. Dans ce cas, l'inverse $(X'X)^{-1}$ n'existe plus et il n'existe plus de solution unique au problème des MCO (4.1).
Une autre situation simple (dans le cas où $n \geq p$) est le cas où les variables $\{X_1, \ldots, X_p\}$ forment une famille liée de $\mathbb{R}^n$. Dans ce cas une (ou plusieurs) variable(s) sont linéairement dépendante(s), c'est-à-dire,

$$\exists j \quad : \quad X_j = \sum_{i \neq j} \alpha_i X_i.$$

Lorsque $rang(X) < p$, la matrice $(X'X)$ n'est pas inversible et la relation donnant $\hat{\beta}$ n'a plus de sens. Nous pouvons cependant toujours projeter Y sur $\mathcal{M}(X)$ mais $\hat{Y}$ n'admet plus une décomposition unique sur les colonnes de X (fig. 4.1) comme nous pouvons l'observer sur le graphique suivant. Le modèle n'est alors pas identifiable.

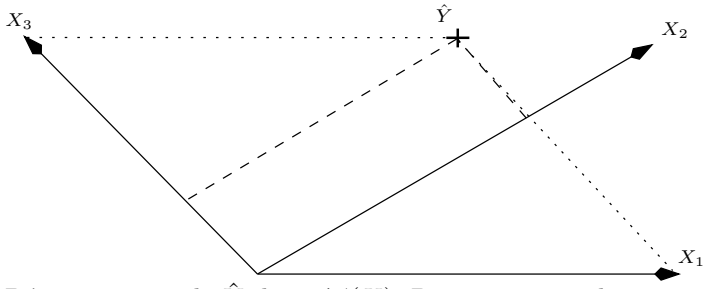


Fig. 4.1 – Décomposition de $\hat{Y}$ dans $\mathcal{M}(X)$. Dans cet exemple nous avons $\hat{Y} = 1 \times X_1 + 1 \times X_3 = 2/3 \times X_2 + 1/3 X_3$.

Remarquons enfin que lorsque X est de plein rang, la variance de $\hat{\beta}$ vaut

$$\mathrm{V}(\hat{\beta}) \quad = \quad \sigma^2 (X'X)^{-1}.$$

et dépend directement du rang de X. Ainsi, même lorsque X est de plein rang mais que $X_j \approx \sum_{i \neq j} \alpha_i X_i$ (on dit souvent « les variables (explicatives) sont très corrélées (empiriquement [1]) »), la variance des estimateurs aura tendance à être élevée et la précision va diminuer. Il est donc important d'utiliser des méthodes adaptées à la déficience de rang. Une méthode assez ancienne pour rendre une matrice inversible consiste à modifier sa diagonale.

4.1.1 Une solution historique

Cette méthode a été proposée par Hoerl & Kennard (1970) et consiste à remplacer $(X'X)^{-1}$ par $(X'X + \lambda I)^{-1}$. Nous obtenons alors l'estimateur ridge

$$\hat{\beta}_{\text{ridge}}(\lambda) = (X'X + \lambda I)^{-1} X'Y,$$

1. Il ne s'agit pas à proprement parler de corrélation, puisque la corrélation empirique simple ne concerne que deux variables (voir exercice 4.2).

où λ est une constante positive à déterminer. Le choix de λ est très important pour la performance de la méthode. En effet,
— $\hat{\beta}_{\text{ridge}}(\lambda) \approx 0$ pour des valeurs de λ élevées ;
— $\hat{\beta}_{\text{ridge}}(\lambda) \approx \hat{\beta}$ pour de faibles valeurs de λ et dans le cas où $\hat{\beta}$ existe.

4.1.2 Minimisation des MCO pénalisés

Nous avons vu dans la partie précédente que la présence de colinéarité entre les colonnes de X a tendance à faire augmenter la variance des estimateurs. Une approche permettant de diminuer cette variance consiste pénaliser le critère des MCO par la norme des paramètres. On est alors amené à minimiser

$$\|Y - X\beta\|^2 + \lambda\|\beta\|^2. \tag{4.2}$$

où $\lambda \geq 0$ est un paramètre à calibrer. D'autres pénalisations sont possibles et cela sera traité en détail au chapitre 8. Pour obtenir la solution de ce problème, nous dérivons par rapport à β

$$2(-X')(Y - X\beta) + 2\lambda\beta.$$

Et en annulant la dérivée, nous obtenons comme solution

$$\hat{\beta}_{\text{ridge}}(\lambda) = (X'X + \lambda I)^{-1}X'Y.$$

On remarque donc que la solution du problème (4.2) est l'estimateur ridge proposé par Hoerl & Kennard (1970).

4.1.3 Equivalence avec une contrainte sur la norme des coefficients

Une autre façon de voir la méthode ridge consiste minimiser le critère des MCO sous une contrainte portant sur la norme des paramètres. On considère l'estimateur $\tilde{\beta}$ défini par

$$\tilde{\beta} = \underset{\beta \in \mathbb{R}^p, \|\beta\|^2 \leq \delta}{\operatorname{argmin}} \|Y - X\beta\|^2, \tag{4.3}$$

où $\delta \geq 0$ est un paramètre à calibrer. Nous calculons le Lagrangien pour obtenir la solution du problème

$$\|Y - X\beta\|^2 + \mu(\|\beta\|^2 - \delta).$$

Une condition nécessaire d'optimum est donnée par l'annulation de ses dérivées partielles au point optimum $(\tilde{\beta}, \tilde{\mu})$:

$$\begin{aligned} -2X'(Y - X\tilde{\beta}) + 2\tilde{\mu}\tilde{\beta} &= 0 \\ \|\tilde{\beta}\|^2 - \delta &= 0. \end{aligned} \tag{4.4}$$

La première équation montre que la solution de ce problème est l'estimateur ridge

$$\tilde{\beta} = \hat{\beta}_{\text{ridge}} = (X'X + \tilde{\mu}I)^{-1}X'Y.$$

Afin de calculer la valeur de $\tilde{\mu}$, pré-multiplions (4.4) à gauche par $\hat{\beta}'_{\text{ridge}}$, cela donne $\tilde{\mu} = (\hat{\beta}'_{\text{ridge}}X'Y - \hat{\beta}'_{\text{ridge}}X'X\hat{\beta}_{\text{ridge}})/\|\hat{\beta}_{\text{ridge}}\|^2$. On peut également vérifier que ce couple est bien un minimum de la fonction en remarquant que le hessien [2] est bien une matrice symétrique de la forme $A'A$, donc semi-définie positive.
Géométriquement, la régression ridge revient à chercher dans une boule de $\mathbb{R}^p$ de rayon δ, le coefficient $\hat{\beta}_{\text{ridge}}$ le plus proche au sens des moindres carrés. En nous plaçant maintenant dans l'espace des observations $\mathbb{R}^n$, l'image de la sphère de contrainte par X est un ellipsoïde de contrainte. Puisque l'ellipsoïde est inclus dans $\Im(X)$, dans le cas où δ est « petit », le coefficient optimum $\hat{\beta}_{\text{ridge}}$ est tel que $X\hat{\beta}_{\text{ridge}}$ est la projection de $X\hat{\beta}$ sur cet ellipsoïde de contrainte (voir fig. 4.2). Dans le cas contraire où $\|\hat{\beta}\|^2 \leq \delta$, $\hat{\beta}$ est dans ou sur l'ellipsoïde et donc sa projection reste égale à $\hat{\beta}$.

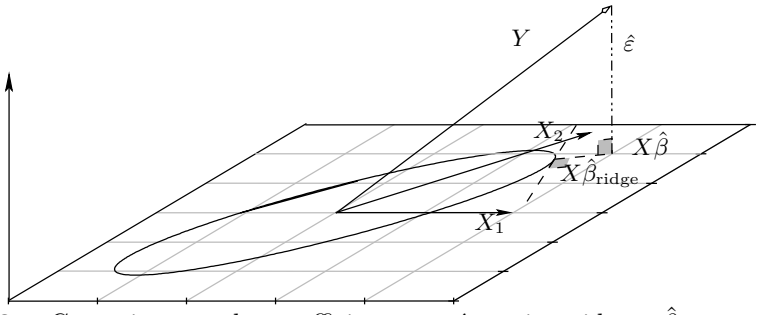


Fig. 4.2 – Contrainte sur les coefficients et régression ridge : $\hat{\beta}_{\text{ridge}}$ représente l'estimateur ridge et $\hat{\beta}$ représente l'estimateur des MC.

Le problème de régression sous contrainte de norme sera abordé plus en détails dans le chapitre 8.

4.1.4 Propriétés statistiques de l'estimateur ridge $\hat{\beta}_{\text{ridge}}$

Revenons aux définitions des estimateurs ridge et MC :

$$\begin{aligned} \hat{\beta}_{\text{ridge}} &= (X'X + \lambda I)^{-1}X'Y \\ \hat{\beta} &= (X'X)^{-1}X'Y. \end{aligned}$$

En pré-multipliant la seconde égalité à gauche par $X'X$, nous avons $X'X\hat{\beta} = X'Y$, cela donne alors

$$\hat{\beta}_{\text{ridge}} = (X'X + \lambda I)^{-1}X'X\hat{\beta}.$$

2. Matrice des dérivées secondes de la fonction.

Cette écriture permet de calculer facilement les propriétés de biais et de variance de l'estimateur ridge. Le calcul de l'espérance de l'estimateur ridge donne

$$\begin{aligned}
\mathbb{E}(\hat{\beta}_{\text{ridge}}) &= (X'X+\lambda I)^{-1}(X'X)\mathbb{E}(\hat{\beta}) \\
&= (X'X+\lambda I)^{-1}(X'X)\beta \\
&= (X'X+\lambda I)^{-1}(X'X+\lambda I-\lambda I)\beta \\
&= \beta-\lambda(X'X+\lambda I)^{-1}\beta.
\end{aligned}$$

Le biais de l'estimateur ridge vaut donc

$$B(\hat{\beta}_{\text{ridge}}) = -\lambda(X'X+\lambda I)^{-1}\beta. \tag{4.5}$$

En général cette quantité est non nulle, l'estimateur ridge est biaisé et la régression est dite biaisée. Calculons la variance de l'estimateur ridge :

$$\begin{aligned}
V(\hat{\beta}_{\text{ridge}}) &= \mathrm{V}((X'X+\lambda I)^{-1}X'Y) \\
&= (X'X+\lambda I)^{-1}X'\,\mathrm{V}(Y)X(X'X+\lambda I)^{-1} \\
&= \sigma^2(X'X+\lambda I)^{-1}X'X(X'X+\lambda I)^{-1}. \qquad (4.6)
\end{aligned}$$

L'estimateur ridge est biaisé, ce qui constitue un handicap par rapport à l'estimateur des MC. En revanche, sa variance fait intervenir $(X'X+\lambda I)^{-1}$ et non pas $(X'X)^{-1}$. Or l'introduction de λI permet d'augmenter les valeurs propres de $(X'X+\lambda I)$ et donc de diminuer la variance.
Après avoir calculé le biais et la variance de cet estimateur, nous allons calculer son EQM (voir p. 39) et le comparer à celui de l'estimateur des MC :

$$\begin{aligned}
\mathrm{EQM}(\hat{\beta}) &= \sigma^2(X'X)^{-1} \\
\mathrm{EQM}(\hat{\beta}_{\text{ridge}}) &= \mathbb{E}(\hat{\beta}_{\text{ridge}}-\beta)\mathbb{E}(\hat{\beta}_{\text{ridge}}-\beta)'+V(\hat{\beta}_{\text{ridge}}) \\
&= \lambda^2(X'X+\lambda I)^{-1}\beta\beta'(X'X+\lambda I)^{-1} \\
&\qquad +\sigma^2(X'X+\lambda I)^{-1}X'X(X'X+\lambda I)^{-1} \\
&= (X'X+\lambda I)^{-1}\left[\lambda^2\beta\beta'+\sigma^2(X'X)\right](X'X+\lambda I)^{-1}.
\end{aligned}$$

Il est difficile de comparer deux matrices, aussi nous prendrons une mesure de la qualité globale *via* la trace. Lorsque nous considérons la trace de la matrice de l'EQM, nous avons

$$\mathrm{tr}[\mathrm{EQM}(\hat{\beta})] = \sigma^2\,\mathrm{tr}((X'X)^{-1}) = \sigma^2\left(\sum_{j=1}^{p}\frac{1}{\lambda_j}\right),$$

où $\{\lambda_j\}_{j=1}^{p}$ sont les valeurs propres de $X'X$. Comme certaines de ces valeurs propres sont nulles ou presque nulles, la trace de l'EQM est donc infinie ou très grande. Nous pouvons montrer que la trace de la matrice de l'EQM de l'estimateur ridge (voir exercice 4.3) est égale à

$$\mathrm{tr}[\mathrm{EQM}(\hat{\beta}_{\text{ridge}})] = \sum_{i=1}^{p}\frac{\sigma^2\lambda_i+\lambda^2[P'\beta]_i^2}{(\lambda_i+\lambda)^2} \quad \text{où} \quad X'X = P\,\mathrm{diag}(\lambda_i)P'.$$

Cette dernière équation donne la forme de l'EQM en fonction du paramètre de la régression ridge λ. Nous pouvons trouver une condition suffisante sur λ (voir exercice 4.3), condition indépendante des variables explicatives,

$$\lambda \leq \frac{2\sigma^2}{\beta'\beta},$$

qui permet de savoir que la trace de l'EQM de l'estimateur ridge est plus petite que celle de l'estimateur des MC. Autrement dit, quand $\lambda \leq 2\sigma^2/\beta'\beta$, la régression ridge est plus précise (dans l'estimation des paramètres) que la régression ordinaire, au sens de la trace de l'EQM. Cependant, cette condition dépend de paramètres inconnus β et σ^2 et elle n'est donc pas utilisable pour choisir une valeur de λ.
Remarquons que si X est orthogonale alors par définition $X'X = I$ et donc la définition de la régression ridge revient à diviser $\hat{\beta} = X'Y$ l'estimateur des MCO par $(1+\lambda)$ et donc à « diminuer » les coefficients d'une même valeur, à l'image de l'estimateur de James-Stein (8.3.3 p. 205).
Nous n'aborderons pas la mise en pratique de la régression ridge et du choix important du paramètre λ, cela sera fait dans le chapitre dédié à la réduction de dimension 8.

4.2 Erreurs corrélées : moindres carrés généralisés

Il existe des cas fréquents où l'hypothèse $\mathcal{H}_2$ (variance des erreurs constante et erreurs non corrélées) n'est pas satisfaite. Les cas rencontrés dans la pratique peuvent être regroupés en deux catégories :

1. La variance des erreurs n'est pas constante, la matrice de variance de ε reste diagonale mais les termes de la diagonale sont différents les uns des autres, on parle alors d'hétéroscédasticité par opposition au cas classique d'homoscédasticité où la variance des erreurs est identique et égale à σ^2.

2. Les erreurs sont corrélées entre elles, la matrice de variance de ε n'est plus diagonale.

Notons la matrice de variance-covariance des erreurs $\Sigma_\varepsilon = \sigma^2\Omega$. Cette matrice Ω est symétrique définie positive [3] et de rang n. Nous allons tout d'abord analyser, en supposant Ω connue, l'impact de cette modification sur les propriétés des estimateurs des MC. L'estimateur des MC est toujours défini par $\hat{\beta} = (X'X)^{-1}X'Y$ et reste sans biais

$$\mathbb{E}(\hat{\beta}) = (X'X)^{-1}X'\mathbb{E}(Y) = \beta,$$

mais sa variance a changé et vaut

$$\mathrm{V}(\hat{\beta}) = (X'X)^{-1}X'\,\mathrm{V}(Y)X(X'X)^{-1} = \sigma^2(X'X)^{-1}X'\Omega X(X'X)^{-1}$$

3. Une matrice de variance-covariance est toujours définie positive.

et dépend donc de Ω. L'estimateur n'est plus de variance minimale parmi les estimateurs linéaires sans biais. En ce qui concerne l'estimateur $\hat{\sigma}^2$ de σ^2, son biais dépend aussi de Ω comme le montrent les calculs suivants :

$$\frac{1}{n-p}\mathbb{E}(\varepsilon' P_X^{\perp}\varepsilon) = \frac{1}{n-p}\mathbb{E}(\mathrm{tr}(P_X^{\perp}\varepsilon\varepsilon')) = \frac{1}{n-p}\,\mathrm{tr}(P_X^{\perp}\Sigma_\varepsilon) = \frac{\sigma^2}{n-p}\,\mathrm{tr}(P_X^{\perp}\Omega).$$

L'estimateur $\hat{\sigma}^2$ ne semble pas adapté puisqu'il est biaisé.
Au cours de cette section, nous allons construire des estimateurs adaptés au problème. Dans un premier temps, nous allons nous intéresser au cas le plus simple, celui de l'hétéroscédasticité et obtenir un estimateur par moindres carrés pondérés. Nous généraliserons ensuite au cas où Ω est définie positive, donnant ainsi la méthode des moindres carrés généralisés.

4.2.1 Erreurs hétéroscédastiques

Considérons donc le modèle

$$Y = X\beta + \varepsilon, \quad \mathbb{E}(\varepsilon) = 0 \quad \text{et} \quad \mathrm{V}(\varepsilon) = \sigma^2\Omega = \sigma^2\,\mathrm{diag}(\omega_1^2, \cdots, \omega_n^2).$$

Une ligne de cette écriture matricielle s'écrit alors

$$y_i = \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i.$$

Une méthode pour obtenir un estimateur sans biais de variance minimale consiste à se ramener à $\mathcal{H}_2$ et à utiliser l'estimateur des MC. Il faudrait donc avoir une variance des résidus constante. En divisant chaque ligne par ω_i nous obtenons

$$\begin{array}{rcl} \dfrac{y_i}{\omega_i} & = & \beta_1\dfrac{x_{i1}}{\omega_i} + \cdots + \beta_p\dfrac{x_{ip}}{\omega_i} + \dfrac{\varepsilon_i}{\omega_i} \\ y_i^* & = & \beta_1 x_{i1}^* + \cdots + \beta_p x_{ip}^* + \varepsilon_i^*. \end{array}$$

La variance de ε^* est constante et vaut σ^2. Nous pouvons donc appliquer les moindres carrés ordinaires sur les variables transformées. Nous obtiendrons un estimateur linéaire sans biais de variance minimale.
Ecrivons cette transformation en écriture matricielle. Définissons $\Omega^{1/2}$ la matrice diagonale des racines carrées des éléments de Ω. Nous avons bien évidemment $\Omega^{1/2}\Omega^{1/2} = \Omega$. L'inverse de la matrice $\Omega^{1/2}$ est une matrice diagonale dont les termes diagonaux sont les inverses des termes diagonaux de $\Omega^{1/2}$, nous noterons cette matrice $\Omega^{-1/2}$, c'est-à-dire

$$\Omega = \begin{pmatrix} \omega_1^2 & & \\ & \ddots & \\ & & \omega_n^2 \end{pmatrix} \quad \text{et} \quad \Omega^{-1/2} = \begin{pmatrix} \frac{1}{\omega_1} & & \\ & \ddots & \\ & & \frac{1}{\omega_n} \end{pmatrix}.$$

L'écriture matricielle de la transformation proposée ci-dessus est donc

$$\begin{array}{rcl} \Omega^{-1/2}Y & = & \Omega^{-1/2}X\beta + \Omega^{-1/2}\varepsilon \\ Y^* & = & X^*\beta + \varepsilon^*. \end{array}$$

Afin de simplifier certaines explications, nous nous référerons à cette modélisation sous le terme « modèle (*) ». La variance de ε^* vaut

$$\mathrm{V}(\varepsilon^*) = \sigma^2\Omega^{-1/2}\Omega\Omega^{-1/2} = \sigma^2\Omega^{-1/2}\Omega^{1/2}\Omega^{1/2}\Omega^{-1/2} = \sigma^2 I_n.$$

Les hypothèses classiques sont vérifiées, nous pouvons estimer β par la méthode des moindres carrés, nous obtenons

$$\hat{\beta}^*_\Omega = (X^{*\prime}X^*)^{-1}X^{*\prime}Y^* = (X'\Omega^{-1}X)^{-1}X'\Omega^{-1}Y.$$

Théorème 4.1 (Gauss-Markov)
L'estimateur $\hat{\beta}^*_\Omega$ est sans biais de variance $\sigma^2(X'\Omega^{-1}X)^{-1}$ minimale.

Nous démontrerons ce théorème dans la partie suivante.

Les valeurs ajustées $\hat{Y}$ sont obtenues par

$$\hat{Y} = X\hat{\beta}_\Omega.$$

Les résidus valent donc

$$\hat{\varepsilon} = Y - \hat{Y}.$$

Remarque
En pratique, il est impossible d'utiliser cette méthode sans connaître les $\{\omega_i\}$. En effet, pour passer au modèle (*), nous supposons les $\{\omega_i\}$ connus. Si les ω_i sont inconnus, nous allons devoir les estimer ainsi que les p paramètres inconnus du modèle. Il est impossible d'estimer $n+p$ paramètres avec n observations. Il existe cependant deux cas pratiques classiques où cette méthode prend tout son sens.

Cas pratique 1 : régression sur données agrégées par groupes

Supposons que les données individuelles suivent le modèle classique de régression

$$Y = X\beta + \varepsilon \quad \mathbb{E}(\varepsilon) = 0 \quad \mathrm{V}(\varepsilon) = \sigma^2\mathrm{I}_n.$$

Cependant, ces données ne sont pas disponibles et nous disposons seulement de moyennes de groupes d'observations : moyenne d'un site, moyenne de différents groupes ou autre... Suite à cette partition en I classes (notées $C_1, \ldots, C_I$) d'effectifs $n_1, \cdots, n_I$ avec $n_1 + \ldots + n_I = n$, nous observons les moyennes par classe :

$$\bar{y}_j = \frac{1}{n_j}\sum_{i\in C_j} y_i, \quad \bar{x}_{jl} = \frac{1}{n_j}\sum_{i\in C_j} x_{il}.$$

Bien évidemment, nous n'observons pas les résidus, mais nous noterons

$$\bar{\varepsilon}_j = \frac{1}{n_j}\sum_{i\in C_j} \varepsilon_i.$$

Le modèle devient alors

$$\bar{Y} = \bar{X}\beta + \bar{\varepsilon} \quad \mathbb{E}(\bar{\varepsilon}) = 0 \quad \mathrm{V}(\bar{\varepsilon}) = \sigma^2\Omega = \sigma^2 \operatorname{diag}(\frac{1}{n_1}, \cdots, \frac{1}{n_I}).$$

Les résultats précédents donnent

$$\hat{\beta}_\Omega = (\bar{X}'\Omega^{-1}\bar{X})^{-1}\bar{X}'\Omega^{-1}\bar{Y}.$$

Lorsque les données sont agrégées par groupes, il est toujours possible d'utiliser l'estimateur des MC. Cependant, cet estimateur n'est pas de variance minimale et l'estimateur de σ^2 obtenu est en général biaisé. Il faut donc utiliser les moindres carrés pondérés et leur estimateur ci-dessus. Les logiciels ne permettent pas toujours de modifier la matrice de variance-covariance des erreurs, l'objectif de ce second cas pratique est de montrer le lien entre hétéroscédasticité et régression pondérée. La régression pondérée est implémentée dans la plupart des logiciels de statistiques.

Cas pratique 2 : régression pondérée

Nous connaissons ici $\Omega = \operatorname{diag}(\omega_1^2, \omega_2^2, \ldots, \omega_n^2)$. Nous venons de voir que, si nous travaillons dans le modèle (*), nous pouvons appliquer les MC classiques. Le problème de minimisation est donc

$$\begin{aligned}
S(\beta) &= \min \sum_{i=1}^{n} \left(y_i^* - \sum_{j=1}^{p} \beta_j x_{ij}^* \right)^2 \\
&= \min \sum_{i=1}^{n} \left(\frac{y_i}{w_i} - \sum_{j=1}^{p} \beta_j \frac{x_{ij}}{w_i} \right)^2 \\
&= \min \sum_{i=1}^{n} \frac{1}{w_i^2} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \\
&= \min \sum_{i=1}^{n} p_i \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .
\end{aligned}$$

Les p_i sont appelés poids et dans les logiciels ces poids sont en général nommés *weight*. Il suffit donc de remplacer les poids par les $1/w_i^2$ et d'appliquer le programme de minimisation pour obtenir l'estimateur

$$\hat{\beta}_\Omega = (X'\Omega^{-1}X)^{-1}X'\Omega^{-1}Y,$$

où

$$\Omega = \operatorname{diag}\left(\omega_1^2, \cdots, \omega_n^2\right) = \operatorname{diag}\left(\frac{1}{p_1}, \cdots, \frac{1}{p_n}\right).$$

4.2.2 Estimateur des moindres carrés généralisés

Nous supposons dans cette partie que le modèle est

$$Y = X\beta + \varepsilon \tag{4.7}$$

et que les hypothèses suivantes sont vérifiées :
$\mathcal{H}_1$: $\text{rang}(X) = p$,
$\mathcal{H}_2'$: $\mathbb{E}(\varepsilon) = 0, \quad \text{V}(\varepsilon) = \sigma^2\,\Omega$, avec $\text{rang}(\Omega) = n$.
L'hypothèse classique $\mathcal{H}_2$ des MC a été modifiée en $\mathcal{H}_2'$. Afin de démontrer aisément pour les estimateurs des moindres carrés généralisés (MCG) toutes les propriétés obtenues pour les estimateurs des MC, à savoir la formule de l'estimateur, son espérance, sa variance, nous allons proposer un changement de variables.
La matrice Ω est symétrique définie positive, il existe donc une matrice P inversible de rang n telle que $\Omega = PP'$. Cette matrice P n'est pas unique. En effet, prenons par exemple une matrice orthogonale Q et utilisons $Z = PQ$ qui vérifie $\Omega = ZZ'$ car $PP' = PQQ'P' = ZZ'$. Le choix de P n'intervient pas dans les résultats qui suivent. Posons $Y^* = P^{-1}Y$ et multiplions à gauche par P^{-1} l'équation (4.7) :

$$\begin{aligned} P^{-1}Y &= P^{-1}X\beta + P^{-1}\varepsilon \\ Y^* &= X^*\beta + \varepsilon^*, \end{aligned}$$

où $X^* = P^{-1}X$ et $\varepsilon^* = P^{-1}\varepsilon$. Dans ce nouveau modèle appelé modèle (*), l'hypothèse concernant le rang de X^* est conservée, $\text{rang}(X^*) = p$. Les hypothèses d'espérance et de variance du bruit ε^* deviennent

$$\begin{aligned} \mathbb{E}(\varepsilon^*) &= 0 \\ \text{V}(\varepsilon^*) &= \text{V}(P^{-1}\varepsilon) = \sigma^2 P^{-1}\Omega P = \sigma^2 I. \end{aligned}$$

Le modèle (*) est donc un modèle linéaire qui satisfait les hypothèses des MC. Pour obtenir toutes les propriétés souhaitées sur le modèle des MCG, il suffira donc d'utiliser les propriétés du modèle (*) et de remplacer X^* par $P^{-1}X$ et Y^* par $P^{-1}Y$.

Estimateur des MCG et optimalité

Ainsi, l'estimateur des MC du modèle (*) vaut

$$\hat{\beta} = (X^{*\prime}X^*)^{-1}X^{*\prime}Y^*,$$

donnant l'estimateur des MCG

$$\hat{\beta}_{MCG} = (X^{*'}X^*)^{-1}X^{*\prime}Y^* = (X'\Omega^{-1}X)^{-1}X'\Omega^{-1}Y.$$

Nous avons donc la définition suivante :

Définition 4.1
L'estimateur des MCG (ou estimateur d'Aitken) est

$$\hat{\beta}_{MCG} = (X'\Omega^{-1}X)^{-1}X'\Omega^{-1}Y.$$

Remarques

— Nous pouvons réinterpréter l'estimateur des MCG avec la notion de métrique particulière de $\mathbb{R}^n$. En effet, il existe une multitude de produits scalaires dans $\mathbb{R}^n$, chacun issu d'une matrice symétrique définie positive M, grâce à

$$\langle u, v\rangle_M = u'Mv.$$

Avec cette remarque, l'estimateur des MCG peut être défini comme le vecteur de $\mathbb{R}^p$ qui minimise la norme $\|Y - X\alpha\|_{\Omega^{-1}}$, définie au sens de la métrique Ω^{-1}. Donc ce vecteur $\hat{\beta}_{MCG}$ est tel que $P_X Y = X\hat{\beta}_{MCG}$, où $P_X = X(X'\Omega^{-1}X)^{-1}X'\Omega^{-1}$ est le projecteur Ω^{-1}-orthogonal sur $\Im(X)$. Il est bien sûr possible de retrouver ce résultat par le calcul en considérant l'orthogonalité entre $Y - X\hat{\beta}_{MCG}$ et un élément de $\Im(X)$. Pour tout vecteur $\alpha \in \mathbb{R}^p$, nous avons

$$\begin{aligned} \langle X\alpha, Y - X\hat{\beta}_{MCG}\rangle_{\Omega^{-1}} &= 0 \\ \alpha' X'\Omega^{-1}(Y - X\hat{\beta}_{MCG}) &= 0, \end{aligned}$$

d'où le résultat.

— Il est possible d'utiliser comme matrice P la matrice $\Omega^{1/2}$ définie par $U\Lambda^{1/2}V'$ où $U\Lambda V'$ est la décomposition en valeurs singulières de Ω.

Les propriétés concernant l'espérance, la variance de l'estimateur des MCG, *i.e.* le théorème de Gauss-Markov, peuvent être déduites du modèle (*) et conduisent au théorème suivant dont la preuve est laissée à titre d'exercice (voir exercice 4.5).

Théorème 4.2 (Gauss-Markov)
L'estimateur $\hat{\beta}_{MCG}$ est sans biais de variance $\sigma^2(X'\Omega^{-1}X)^{-1}$ et meilleur que tout estimateur linéaire sans biais, au sens où sa variance est minimale.

Sous l'hypothèse $\mathcal{H}_2'$, l'estimateur des MC, $\hat{\beta}_{MC} = (X'X)^{-1}X'Y$, est toujours linéaire en Y et sans biais, mais n'est plus de variance minimale.

Résidus et estimateur de σ^2

Les résidus sont définis par $\hat{\varepsilon} = Y - X\hat{\beta}_{MCG}$. Remarquons qu'à l'image du vrai bruit où nous avons $\varepsilon^* = P^{-1}\varepsilon$, nous avons pour l'estimation $\hat{\varepsilon}^* = P^{-1}\hat{\varepsilon}$.
Un estimateur de σ^2 est donné par

$$\hat{\sigma}^2_{MCG} = \frac{\|Y - X\hat{\beta}_{MCG}\|^2_{\Omega^{-1}}}{n-p}.$$

Proposition 4.1
L'estimateur $\hat{\sigma}^2_{MCG}$ est un estimateur sans biais de σ^2.

Preuve

$$\begin{aligned}
(n-p)\hat{\sigma}^2_{MCG} &= \langle Y - X\hat{\beta}_{MCG}, Y - X\hat{\beta}_{MCG}\rangle_{\Omega^{-1}} \\
&= (Y - X\hat{\beta}_{MCG})'\Omega^{-1}(Y - X\hat{\beta}_{MCG}) \\
&= (PP^{-1}(Y - X\hat{\beta}_{MCG}))'\Omega^{-1}(PP^{-1}(Y - X\hat{\beta}_{MCG})) \\
&= (Y^* - X^*\hat{\beta}_{MCG})'P'\Omega^{-1}P(Y^* - X^*\hat{\beta}_{MCG}) \\
&= \hat{\varepsilon}^{*'}\hat{\varepsilon}^*.
\end{aligned}$$

Dans le modèle (*), $\hat{\sigma}^2_{MCG}$ est un estimateur sans biais de σ^2, d'où le résultat. □

4.2.3 Matrice Ω inconnue

Dans les problèmes rencontrés, la matrice Ω est souvent inconnue. Il faut donc l'estimer puis remplacer dans les calculs Ω par son estimateur $\hat{\Omega}$. Cependant, si nous n'avons aucune information sur Ω, il est impossible d'estimer les termes de Ω car il faut estimer $(n^2 - n)/2$ termes non diagonaux et n termes diagonaux. Il est cependant possible d'estimer Ω dans certains cas particuliers :

— Ω diagonale de forme particulière (voir 4.2.1, p. 80) ;

— Ω admet une expression particulière paramétrable avec seulement quelques paramètres (σ^2, θ) à estimer.

En règle générale, pour estimer θ, on maximise la vraisemblance $\mathcal{L}(\beta, \sigma^2, \theta)$. Cependant, nous allons détailler un premier exemple classique où l'estimation de θ est conduite par une procédure beaucoup plus simple.

Corrélation temporelle Considérons le modèle

$$Y = X\beta + \varepsilon$$

où l'erreur est supposée suivre un processus autorégressif $\varepsilon_t = \rho\varepsilon_{t-1} + \eta_t$ avec $0 < \rho < 1$ et où $\mathrm{Cov}(\eta_i, \eta_j) = \sigma^2_\varepsilon \delta_{ij}$. Notons que $\sigma^2_\eta = \sigma^2_\varepsilon(1 - \rho^2)$. La matrice (symétrique) de variance Ω des erreurs ε, fonction de deux paramètres inconnus σ^2_ε et ρ, s'écrit alors

$$\Omega = \frac{\sigma^2_\eta}{1-\rho^2}\begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{n-1} \\ & 1 & \rho & \cdots & \rho^{n-2} \\ & & \ddots & & \\ & & & \ddots & \\ & & & & 1 \end{pmatrix}.$$

Le calcul de son inverse donne

$$\Omega^{-1} = \begin{pmatrix} 1 & -\rho & 0 & \cdots & \cdots & 0 \\ & 1+\rho^2 & -\rho & 0 & \cdots & 0 \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & 0 \\ & & & & 1+\rho^2 & -\rho \\ & & & & & 1 \end{pmatrix}.$$

Nous venons de calculer Ω^{-1} dans ce cas précis. Afin de calculer l'estimateur d'Aitken de β, il faut estimer Ω^{-1} et donc estimer ρ. Pour pouvoir estimer ρ, il faudrait disposer des ε_t et ce n'est évidemment pas le cas.
Dans la pratique, nous calculons $\hat{\beta}_{MC} = (X'X)^{-1}X'Y$, et calculons les résidus $\hat{\varepsilon} = Y - X\hat{\beta}_{MC}$. Nous supposons ensuite que $\hat{\varepsilon}_t = \rho\hat{\varepsilon}_{t-1} + \eta_t$, nous pouvons donc estimer ρ par les MC, cela nous donne

$$\hat{\rho}_{MC} = \frac{\sum_{t=2}^{n} \hat{\varepsilon}_t \hat{\varepsilon}_{t-1}}{\sum_{t=2}^{n} \hat{\varepsilon}_{t-1}^2}.$$

A partir de cet estimateur, nous estimons Ω par $\hat{\Omega}$ puis appliquons l'estimateur d'Aitken :

$$\hat{\beta}_{MCG} = (X'\hat{\Omega}^{-1}X)^{-1}X'\hat{\Omega}^{-1}Y.$$

Cet estimateur a été calculé en deux étapes (*two stages*), estimation des résidus par MC puis, à partir des résidus estimés, calcul de $\hat{\beta}_{MCG}$. Cet estimateur est appelé $\hat{\beta}_{TS}$ pour *two stages*. Un autre estimateur peut être trouvé en itérant ce processus jusqu'à convergence, l'estimateur est alors qualifié d'itéré (*iterated*).

4.3 Exercices

Exercice 4.1 (Questions de cours)

1) Lorsque la matrice $(X'X)$ n'est pas inversible, l'estimateur des moindres carrés
   A. existe et est unique,
   B. existe et n'est pas unique,
   C. n'existe pas, aucun estimateur ne minimise les moindres carrés.
2) La variance de l'estimateur ridge est
   A. toujours plus grande que la variance de l'estimateur des MCO,
   B. toujours plus petit que la variance de l'estimateur des MCO,
   C. cela dépend de λ.
3) Nous utilisons les MCG car l'hypothèse suivante n'est pas satisfaite :
   A. $\mathcal{H}_1$ le rang du plan d'expérience,
   B. $\mathcal{H}_2$ l'espérance et la variance des résidus,
   C. $\mathcal{H}_3$ la normalité des résidus.
4) La matrice de variance de ε est Ω. Nous calculons $\hat{\beta}_{MCG}$ et $\hat{\beta}_{MC}$. Avons-nous ?
   A. $\mathrm{V}(\hat{\beta}_{MCG}) \leq \mathrm{V}(\hat{\beta}_{MC})$,
   B. $\mathrm{V}(\hat{\beta}_{MCG}) \geq \mathrm{V}(\hat{\beta}_{MC})$,
   C. les variances ne peuvent pas être comparées.

Exercice 4.2 (Corrélation multiple et hypothèse $\mathcal{H}_1$)
Soit Y une variable continue et $X_1, \ldots, X_p$ p variables continues avec $X_1 = \mathbb{1}_n$. Le coefficient de corrélation linéaire multiple entre Y et $X_1, \ldots, X_p$ est défini *par la valeur maximale de la corrélation (empirique) linéaire* $\rho(.)$ *entre* Y *et une combinaison linéaire des variables* $X_1, \ldots, X_p$

$$\mathrm{R}(Y; X) = \mathrm{R}(Y; X_1, \ldots, X_p) = \sup_{\beta \in \mathbb{R}^p} \rho(Y; X\beta).$$

1) Etablir que le R^2 de la régression multiple de Y sur $X_1, \ldots, X_p$ est le carré de $\rho(Y; X\hat{\beta})$ (indice : montrer que la moyenne empirique de $X\hat{\beta}$ vaut $\bar{Y}$).
2) Soit

$$X_1 = \mathbb{1}_3,\ X_2 = (1/\sqrt{2}, 1/\sqrt{2}, -\sqrt{2})' \text{ et}$$
$$Y = \Big(\frac{2(\sqrt{2}-1)+3\sqrt{3}}{\sqrt{2}}, \frac{2(\sqrt{2}-1)-3\sqrt{3}}{\sqrt{2}}, 2(1+\sqrt{2})\Big)'.$$

a) Montrer que $Y = 2X_1 - 2X_2 + 3\eta$, où $\eta = (\sqrt{3}/\sqrt{2}, -\sqrt{3}/\sqrt{2}, 0)'$.
b) Montrer que $\|X_1\| = \|X_2\| = \|\eta\|$ et que $X_1 \perp X_2 \perp \eta$.
c) Trouver $\hat{Y} = P_X Y$. Représenter dans le repère (O, X_1, X_2, η) $\overrightarrow{OY}$ et $\overrightarrow{O\hat{Y}}$.
d) Que représente graphiquement $\rho(Y; X\hat{\beta})$?
e) Que représente graphiquement $\rho(Y; X\alpha)$, avec $\alpha = (4, -3)'$?
f) Déduire graphiquement que $\hat{\beta}$ réalise le maximum de $\sup_{\beta \in \mathbb{R}^2} \rho(Y; X\beta)$.

3) Soit une variable X_j. Notons $X_{(j)}$ la matrice X privée de sa j^{e} colonne. Etablir que si $\mathrm{R}(X_j; X_{(j)}) = 1$, alors $\mathcal{H}_1$ n'est pas vérifiée. En déduire alors que si X_j et X_k sont corrélées linéairement ($\rho(X_j, X_k) = 1$ avec $j \neq k$), alors $\mathcal{H}_1$ n'est pas vérifiée.

Exercice 4.3 (†EQM de la régression ridge)
Soit le modèle habituel de régression

$$Y = X\beta + \varepsilon.$$

1) Donner l'expression de l'estimateur ridge $\hat{\beta}_{\text{ridge}}$ et calculer son biais, sa variance et sa matrice de l'EQM.
2) En utilisant la décomposition en valeurs singulières (ou valeurs propres) de $X'X = P\,\mathrm{diag}(\lambda_j)P'$, établir en utilisant la question 2) de l'exercice 8.4 que

$$\mathrm{tr}(\mathrm{EQM}(\hat{\beta}_{\text{ridge}})) = \sum_{j=1}^{r} \frac{\sigma^2\lambda_j + \kappa^2[P'\beta]_j^2}{(\lambda_j+\kappa)^2}.$$

3) Retrouver que la matrice de l'EQM pour l'estimateur des MC est

$$\begin{aligned}\mathrm{EQM}(\hat{\beta}_{\text{MC}}) &= \sigma^2(X'X)^{-1}\\ &= \sigma^2(X'X+\kappa I)^{-1}(X'X+\kappa^2(X'X)^{-1}+2\kappa I_p)(X'X+\kappa I)^{-1}.\end{aligned}$$

4) Calculer la différence entre la matrice de l'EQM pour l'estimateur ridge et celle pour l'estimateur des MC et montrer l'égalité suivante :

$$\begin{aligned}\Delta &= \mathrm{EQM}(\hat{\beta}_{\text{ridge}}) - \mathrm{EQM}(\hat{\beta}_{\text{MC}})\\ &= \kappa(X'X+\kappa I)^{-1}(\sigma^2(2I_p+\kappa^2(X'X)^{-1}) - \kappa\beta\beta')(X'X+\kappa I)^{-1}.\end{aligned}$$

5) En utilisant la propriété suivante *Si A est inversible, alors une condition nécessaire et suffisante pour que B soit semi-définie positive est que ABA' le soit aussi*, déduire qu'une condition nécessaire et suffisante pour que Δ soit semi-définie positive est que $(\sigma^2(2I_p+\kappa^2(X'X)^{-1}) - \kappa\beta\beta')$ le soit aussi.
6) Démontrer que $I_p - \gamma\gamma'$ est semi-définie positive si et seulement si $\gamma'\gamma \leq 1$ (utiliser la décomposition en valeurs singulières (ou propres) de $\gamma\gamma'$ dont on calculera les valeurs propres et le théorème ci-dessus).
7) En utilisant la propriété suivante *Si A et B sont des matrices semi-définies positives, alors pour tout $\alpha > 0$ et $\beta > 0$ la matrice $\alpha A + \beta B$ est aussi semi-définie positive*, déduire qu'une condition suffisante pour que Δ soit semi-définie positive est que $\kappa \leq 2\sigma^2/\beta'\beta$.

8) Conclure sur la différence des traces des EQM des estimateurs des MC et ridge.

Exercice 4.4 (Régression pondérée)
Nous voulons effectuer une régression pondérée, c'est-à-dire que nous voulons minimiser

$$\hat{\beta}_{pond} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 p_i,$$

où p_i est un réel positif (le poids).

1) Afin de trouver $\hat{\beta}_{pond}$, trouver un changement de variable dans lequel le critère à minimiser s'écrit comme les moindres carrés classiques avec les nouvelles variables $X^\star$ et $Y^\star$.
2) En appliquant le changement de variable précédent, trouver l'estimateur $\hat{\beta}_{pond}$.
3) Montrer que lorsque la seule variable explicative est la constante, la solution est

$$\hat{\beta}_1 = \frac{\sum p_i y_i}{\sum p_i}.$$

4) Retrouver un estimateur connu si les p_i sont constants pour $i = 1, \cdots, n$.

Exercice 4.5 (Gauss-Markov)
Démontrer le Théorème 4.2.

Exercice 4.6 (Corrélation spatiale)
Considérons l'exemple du livre de Upton & Fingleton (1985) : l'objectif est d'expliquer le nombre de plantes endémiques observées par trois variables : la surface de l'unité de mesure, l'altitude et la latitude sur différents sites. Les résidus ε admettent une dépendance entre sites et nous considérons le modèle :

$$Y = X\beta + \varepsilon \quad \text{avec} \quad \varepsilon = \rho M \varepsilon + \eta, \quad \eta \sim \mathcal{N}(0, \sigma^2 I_n), \tag{4.8}$$

où M est une matrice *connue* de dépendance entre sites avec $M_{ii} = 0$ et définie par la distance en miles entre sites grâce à

$$M_{ij} = \frac{D_{ij}}{\sum_{j=1}^{n} D_{ij}}$$

où les termes de la matrice D sont définis par

$$D_{ij} = \begin{cases} \frac{1}{d(i,j)^2} \text{ si } d(i,j) < 187.5 \text{ miles,} \\ 0 \quad \text{si } d(i,j) \geq 187.5 \text{ ou si } i \text{ ou } j \text{ est une île} \end{cases}$$

où $d(i,j)$ est la distance en miles entre le site i et le site j.

1) Montrer qu'en réécrivant cette équation pour un site i, vous avez

$$\varepsilon_i = \rho \sum_{j \neq i, j=1}^{n} M_{ij} \varepsilon_j + \eta_i,$$

2) Interprétez cette écriture. Ce modèle est noté SAR pour *Simultaneous AutoRegressive*.
3) Montrer à partir de (4.8) que

$$\begin{aligned} (I_n - \rho M)\varepsilon &= \eta \\ \varepsilon &= (I_n - \rho M)^{-1}\eta = A^{-1}\eta. \end{aligned}$$

4) La variable η suit une loi normale $\mathcal{N}(0, \sigma^2 I_n)$, quelle est la loi de ε ?
5) Montrer que la variance de η vaut $\sigma^2\Omega = \sigma^2 A^{-1} A'^{-1}$.
6) Montrer que la vraisemblance du modèle s'écrit

$$L(Y, \beta, \sigma^2, \rho) = (2\pi\sigma^2)^{-\frac{n}{2}} |\Omega|^{-\frac{1}{2}} \exp\Big\{-\frac{1}{2\sigma^2}(Y - X\beta)'\Omega^{-1}(Y - X\beta)\Big\}.$$

7) En dérivant la log-vraisemblance et en annulant les dérivées en $(\hat\beta, \hat\sigma^2, \hat\rho)$, montrer que

$$\begin{aligned}\hat\beta &= (X'\hat\Omega^{-1}X)^{-1}X'\hat\Omega^{-1}Y \\ &= (X'\hat A'\hat A X)^{-1}X'\hat A'\hat A Y.\end{aligned}$$

Remarquez que $\hat\beta$ est une fonction de $\hat\rho$ uniquement. Si nous connaissons $\hat\rho$ nous connaissons $\hat\beta$.
8) Montrer que

$$\hat\sigma^2 = \frac{1}{n}(Y - X\hat\beta)\hat A'\hat A(Y - X\hat\beta).$$

Ainsi, une fois estimé ρ par $\hat\rho$, nous pouvons déterminer $\hat\beta$ puis $\hat\sigma$.
9) Déduire des questions précédentes que

$$\mathcal{L}(Y, \hat\beta, \hat\sigma^2, \hat\rho) = -\frac{n}{2}\log\hat\sigma^2 + \frac{1}{2}\log|\hat A'\hat A| + cte$$

10) Montrer que l'opposé de la vraisemblance s'écrit comme une fonction uniquement de $\hat\rho$ (à une constante près) :

$$\begin{aligned}h(\hat\rho) &= \frac{n}{2}\log Y'(I - X(X'\hat A'\hat A X)^{-1}X'\hat A'\hat A)'\hat A'\hat A(I - X(X'\hat A'\hat A X)^{-1}X'\hat A'\hat A)Y \\ &\quad -\frac{1}{2}\log|\hat A'\hat A|^2\end{aligned}$$

Voici le code pour optimiser cette fonction :

```
> n <- nrow(don)
> X <- cbind(rep(1,n),data.matrix(don[,-1]))
> y <- data.matrix(don[,1])
> concentree <- function(rho,MM,nn,yy,XX) {
+   AA <- diag(nn)-rho*MM
+   PP <- AA%*%(diag(nn)-XX%*%
+         solve(crossprod(AA%*%XX))%*%t(XX)%*%crossprod(AA))%*%yy
+   res <- 0.5*nn*log(crossprod(PP))-0.5*(log(det(crossprod(AA))))
+   return(res)
+ }
> resconc <- optimize(concentree,c(-1,1),MM=M,nn=n,yy=y,XX=X)
```

Puis pour estimer les paramètres, vous pouvez utiliser :

```
> rhoconc <- resconc$minimum
> A <- diag(n)-rhoconc*M
> betaconc <- solve(crossprod(A%*%X))%*%t(X)%*%crossprod(A)
+                  %*%(as.matrix(don[,"nbe.plante"]))
> sigmaconc <- sqrt(as.vector(crossprod(A%*%
+               (as.matrix(don[,"nbe.plante"])-X%*%betaconc)))/n)
```

Deuxième partie

Inférence

Chapitre 5

Inférence dans le modèle gaussien

Nous rappelons le contexte du chapitre précédent :

$$Y_{n\times 1} = X_{n\times p}\ \beta_{p\times 1} + \varepsilon_{n\times 1},$$

sous les hypothèses
— $\mathcal{H}_1 : \mathrm{rang}(X) = p$;
— $\mathcal{H}_2 : \mathbb{E}(\varepsilon) = 0, \quad \Sigma_\varepsilon = \sigma^2\, \mathrm{I}_n$.
Nous allons désormais supposer que les erreurs suivent une loi normale, donc $\mathcal{H}_2$ devient
— $\mathcal{H}_3 : \varepsilon \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_n)$.
Nous pouvons remarquer que $\mathcal{H}_3$ implique $\mathcal{H}_2$. Dans le cas gaussien, si $\mathrm{Cov}(\varepsilon_i, \varepsilon_j) = \sigma^2\delta_{ij}$ alors les ε_i sont indépendants. L'hypothèse $\mathcal{H}_3$ s'écrit $\varepsilon_1, \cdots, \varepsilon_n$ sont i.i.d. et de loi $\mathcal{N}(0, \sigma^2)$.
L'hypothèse gaussienne va nous permettre de calculer la vraisemblance et les estimateurs du maximum de vraisemblance (EMV). Cette hypothèse va nous permettre également de calculer des régions de confiance et de proposer des tests. C'est l'objectif de ce chapitre.

5.1 Estimateurs du maximum de vraisemblance

Calculons la vraisemblance de l'échantillon. La vraisemblance est la densité de l'échantillon vue comme fonction des paramètres. Grâce à l'indépendance des erreurs, les observations sont indépendantes et la vraisemblance s'écrit :

$$L(Y, \beta, \sigma^2) = \prod_{i=1}^{n} f_Y(y_i) = \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i - \sum_{j=1}^{p}\beta_j x_{ij}\big)^2\right].$$

Nous avons donc

$$L(Y,\beta,\sigma^2) = \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left[-\frac{1}{2\sigma^2}\|Y - X\beta\|^2\right].$$

Il est souvent plus simple de considérer la log-vraisemblance

$$\mathcal{L}(Y,\beta,\sigma^2) = \log L(Y,\beta,\sigma^2) = -\frac{n}{2}\log\sigma^2 - \frac{n}{2}\log 2\pi - \frac{1}{2\sigma^2}\|Y - X\beta\|^2.$$

Dérivons $\mathcal{L}$ par rapport aux paramètres

$$\frac{\partial\mathcal{L}(Y,\beta,\sigma^2)}{\partial\beta} = \frac{1}{2\sigma^2}\frac{\partial}{\partial\beta}\left(\|Y - X\beta\|^2\right), \tag{5.1}$$

$$\frac{\partial\mathcal{L}(Y,\beta,\sigma^2)}{\partial\sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\|Y - X\beta\|^2. \tag{5.2}$$

Annulons ces dérivées. A partir de (5.1), nous avons évidemment $\hat{\beta}_{MV} = \hat{\beta}$ et à partir de (5.2) nous avons

$$\hat{\sigma}^2_{MV} = \frac{\|Y - X\hat{\beta}_{MV}\|^2}{n}$$

donc $\hat{\sigma}^2_{MV} = (n-p)\hat{\sigma}^2/n$. L'estimateur du MV de σ^2 est donc biaisé par opposition à l'estimateur $\hat{\sigma}^2$ obtenu par les MC. Afin de vérifier que nous avons bien un maximum, il faut étudier les dérivées secondes (à faire en exercice).
Sous l'hypothèse supplémentaire $\mathcal{H}_3$, les propriétés établies au chapitre 2 sont toujours valides (sans biais, variance minimale). Nous pouvons toutefois établir de nouvelles propriétés.

5.2 Nouvelles propriétés statistiques

Grâce à l'hypothèse gaussienne, nous pouvons « améliorer » le théorème de Gauss-Markov. L'optimalité des estimateurs est élargie et nous ne considérons plus seulement les estimateurs linéaires sans biais, mais la classe plus grande des estimateurs sans biais. De plus, le théorème intègre désormais l'estimateur de σ^2. La preuve de ce théorème est donnée parmi les corrections des exercices de ce chapitre.

Proposition 5.1
$(\hat{\beta},\hat{\sigma}^2)$ est une statistique complète et $(\hat{\beta},\hat{\sigma}^2)$ est de variance minimum dans la classe des estimateurs sans biais.

Nous pouvons ensuite établir une proposition importante pour la construction des tests et régions de confiance.

Proposition 5.2 (Lois des estimateurs : variance connue)
Sous les hypothèses $\mathcal{H}_1$ et $\mathcal{H}_3$, nous avons
i) $\hat{\beta}$ est un vecteur gaussien de moyenne β et de variance $\sigma^2(X'X)^{-1}$,
ii) $(n-p)\hat{\sigma}^2/\sigma^2$ suit un χ^2 à $n-p$ ddl (χ^2_{n-p}),
iii) $\hat{\beta}$ et $\hat{\sigma}^2$ sont indépendants.

Preuve
i) $\hat{\beta}$ est fonction linéaire de variables gaussiennes et suit donc une loi normale entièrement caractérisée par son espérance et sa variance calculées au chapitre précédent.
ii)

$$\hat{\sigma}^2 = \frac{\|Y - X\hat{\beta}\|^2}{n-p} = \frac{1}{n-p}\|\hat{\varepsilon}\|^2 = \frac{1}{n-p}\|P_{X^\perp}\varepsilon\|^2 = \frac{1}{n-p}\varepsilon' P_{X^\perp}\varepsilon.$$

Or $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$ et $P_{X^\perp}$ est la matrice de projection orthogonale sur $\Im(X)^\perp$, espace de dimension $n-p$. Nous obtenons le résultat par le théorème de Cochran (théorème A.1 p. 418).
iii) Remarquons que $\hat{\beta}$ est fonction de $P_X Y$ ($\hat{\beta} = (X'X)^{-1}X'P_X Y$) et $\hat{\sigma}^2$ est fonction de $(I - P_X)Y$. Les vecteurs gaussiens $\hat{Y}$ et $\hat{\varepsilon}$ sont de covariance nulle et sont donc indépendants. Toute fonction fixe de $\hat{Y}$ reste indépendante de toute fonction fixe de $\hat{\varepsilon}$, d'où le résultat. □

Il en découle une proposition plus générale pour bâtir les régions de confiance.

Proposition 5.3 (Lois des estimateurs : variance estimée)
Sous les hypothèses $\mathcal{H}_1$ et $\mathcal{H}_3$, nous avons

i) pour $j = 1, \cdots, p$, $T_j = \dfrac{\hat{\beta}_j - \beta_j}{\hat{\sigma}\sqrt{[(X'X)^{-1}]_{jj}}} \sim \mathcal{T}(n-p)$,

ii) soit R une matrice de taille $q \times p$ de rang q ($q \leq p$) alors

$$\frac{1}{q\hat{\sigma}^2}(R(\hat{\beta} - \beta))' \left[R(X'X)^{-1}R'\right]^{-1} R(\hat{\beta} - \beta) \sim \mathcal{F}_{q,n-p}.$$

Preuve
i) la variance de l'estimateur $\hat{\beta}_j$ vaut $\sigma^2[X'X]^{-1}_{jj}$, nous avons alors

$$\frac{\hat{\beta}_j - \beta_j}{\sigma\sqrt{[(X'X)^{-1}]_{jj}}} \sim \mathcal{N}(0,1).$$

σ^2 est inconnue et estimée par $\hat{\sigma}^2$. La suite découle de l'utilisation des points (ii) et (iii) de la proposition précédente.
ii) Le rang de R vaut par hypothèse $q \leq p$, donc le rang de $R(X'X)^{-1}R'$ vaut q. $R\hat{\beta}$ est un vecteur gaussien de moyenne $R\beta$ et de variance $\sigma^2 R(X'X)^{-1}R'$. Nous avons donc

$$\frac{1}{\sigma^2}(R\hat{\beta} - R\beta)' \left[R(X'X)^{-1}R'\right]^{-1} (R\hat{\beta} - R\beta) \sim \chi^2_q. \tag{5.3}$$

Or σ^2 est inconnue. Afin de faire disparaître σ^2 de l'équation (5.3), nous divisons le membre de gauche par $\hat{\sigma}^2/\sigma^2$. Rappelons que par (ii) nous savons que $\hat{\sigma}^2/\sigma^2$ suit un χ^2 divisé par son degré de liberté et que par (iii) $\hat{\sigma}^2/\sigma^2$ est indépendant du membre de gauche de l'équation (5.3). La suite découle donc de la définition d'une loi de Fisher (rapport de deux χ^2 indépendants divisés par leurs degrés de liberté respectifs). □

5.3 Intervalles et régions de confiance

Les logiciels et certains ouvrages donnent des IC pour les paramètres pris séparément. Cependant ces IC ne tiennent pas compte de la dépendance des estimations. Il est possible d'obtenir des IC simultanés pour plusieurs paramètres. Le théorème ci-dessous détaille toutes les formes de RC : simple ou simultané. C'est le théorème central de l'estimation par intervalle dont la démonstration est à faire à titre d'exercice (voir exercice 5.2).

Théorème 5.1 (IC et RC des paramètres)
i) Un IC *bilatéral de niveau* $1-\alpha$*, pour un* β_j *pour* $j = 1, \cdots, p$ *est donné par*

$$\left[\hat{\beta}_j - t_{n-p}(1-\alpha/2)\hat{\sigma}\sqrt{[(X'X)^{-1}]_{jj}}, \quad \hat{\beta}_j + t_{n-p}(1-\alpha/2)\hat{\sigma}\sqrt{[(X'X)^{-1}]_{jj}}\right].$$

ii) Un IC *bilatéral de niveau* $1-\alpha$*, pour* σ^2 *est donné par*

$$\left[\frac{(n-p)\hat{\sigma}^2}{c_2}, \quad \frac{(n-p)\hat{\sigma}^2}{c_1}\right] \quad \text{où} \quad P(c_1 \leq \chi^2_{n-p} \leq c_2) = 1-\alpha.$$

iii) Une RC *pour* q $(q \leq p)$ *paramètres* β_j *notés* $(\beta_{j_1}, \cdots, \beta_{j_q})$ *de niveau* $1-\alpha$ *est donnée,*

- *lorsque* σ *est connue, par*

$$\mathrm{RC}_\alpha(R\beta) = \left\{R\beta \in \mathbb{R}^q, \frac{1}{\sigma^2}[R(\hat{\beta}-\beta)]'[R(X'X)^{-1}R']^{-1}[R(\hat{\beta}-\beta)] \leq \chi^2_q(1-\alpha)\right\}$$

- *lorsque* σ *est inconnue, par*

$$\begin{aligned}\mathrm{RC}_\alpha(R\beta) = \{&R\beta \in \mathbb{R}^q, \\ &\frac{1}{q\hat{\sigma}^2}\,[R(\hat{\beta}-\beta)]'[R(X'X)^{-1}R']^{-1}[R(\hat{\beta}-\beta)] \leq f_{q,n-p}(1-\alpha)\Big\}, \quad (5.4)\end{aligned}$$

  où R *est la matrice de taille* $q \times p$ *dont tous les éléments sont nuls sauf les* $[R]_{ij_i}$ *qui valent 1.*

Les valeurs c_1 *et* c_2 *sont les fractiles d'un* χ^2_q *et* $f_{q,n-p}(1-\alpha)$ *est le fractile de niveau* $(1-\alpha)$ *d'une loi de Fisher admettant* $(q, n-p)$ *ddl.*

Exemple : différence entre intervalles et régions de confiance
Nous souhaitons donner une RC pour β_1 et β_2, la matrice R est donnée par

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \qquad R(\hat{\beta}-\beta) = \begin{bmatrix} \hat{\beta}_1 - \beta_1 \\ \hat{\beta}_2 - \beta_2 \end{bmatrix}.$$

Nous avons alors pour (β_1, β_2) la RC suivante lorsque σ^2 est inconnu :

$$\mathrm{RC}_\alpha(\beta_1, \beta_2) = \left\{\frac{1}{2\hat{\sigma}^2}[\hat{\beta}_1-\beta_1, \hat{\beta}_2-\beta_2][R(X'X)^{-1}R']^{-1}\begin{bmatrix} \hat{\beta}_1 - \beta_1 \\ \hat{\beta}_2 - \beta_2 \end{bmatrix} \leq f_{2,n-p}(1-\alpha)\right\}.$$

Notons c_{ij} le terme général de $(X'X)^{-1}$, nous obtenons en développant

$$\mathrm{RC}_\alpha(\beta_1,\beta_2) = \Big\{(\beta_1,\beta_2)\in\mathbb{R}^2,\quad \frac{1}{2\hat\sigma^2(c_{11}c_{22}-c_{12}^2)}\times$$
$$\Big(c_{22}(\hat\beta_1-\beta_1)^2 - 2c_{12}(\hat\beta_1-\beta_1)(\hat\beta_2-\beta_2) + c_{11}(\hat\beta_2-\beta_2)^2\Big) \le f_{2,n-p}(1-\alpha)\Big\}.$$

Cette région de confiance est une ellipse qui tient compte de la corrélation entre $\hat\beta_i$ et $\hat\beta_j$, contrairement à la juxtaposition de deux intervalles de confiance qui forme un rectangle (voir fig. 5.1).

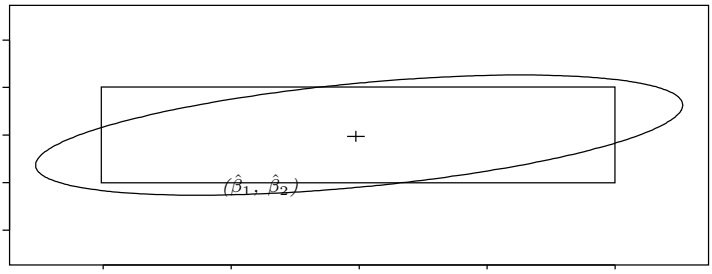


Fig. 5.1 – Comparaison entre ellipse et rectangle de confiance.

Si les composantes sont très peu corrélées alors les régions parallélépipédiques définies par les IC sont une bonne approximation de l'ellipsoïde.
Nous traitons les 50 données journalières concernant la concentration en ozone. La variable à expliquer est la concentration en ozone notée `O3` et les variables explicatives sont la température `T12`, le vent `Vx` et la nébulosité `Ne12`.
Après avoir estimer le modèle, et afin de calculer les intervalles de confiance à 95 % pour les paramètres, il suffit d'utiliser les ordres suivants :

```
> modele3 <- lm(O3 ~ T12 + Vx + Ne12, data = ozone)
> resume3 <- summary(modele3)
> coef3 <- coef(resume3)
> IC3 <- t(confint(modele3, level = 0.95))
> IC3
       (Intercept)       T12        Vx      Ne12
2.5 %     57.15842 0.3138112 0.1491857 -6.960609
97.5 %   111.93625 2.3162807 0.8237055 -2.826137
```

où la fonction `confint` calcule directement la valeur des bornes de l'intervalle grâce au théorème 5.1. Pour vérifier numériquement le théorème, la valeur numérique du quantile $t_{n-p}(1-\alpha/2)$ est obtenue grâce à `qt(0.975,modele3$df.res)` tandis que les valeurs des estimations des variances sont dans `resume3$coefficients[,2]`.
Afin de dessiner les ellipses de confiance, nous utilisons le package **ellipse** : Nous allons dessiner les régions de confiance de tous les couples de paramètres et les com-

parer graphiquement aux intervalles de confiance pour chaque paramètre pris indépendamment (ellipse *versus* rectangle). Nous choisissons un intervalle de confiance à 95 % pour chaque paramètre et une région de confiance à 95 %. Nous obtenons le dessin des ellipses de confiance pour tous les couples de paramètres grâce aux commandes suivantes :

```
> library(ellipse)
> par(mfrow=c(3,2))
> for(i in 1:3){
+  for(j in (i+1):4){
+    plot(ellipse(modele3,c(i,j),level=0.95),type="l",
+     xlab=paste("beta",i,sep=""),ylab=paste("beta",j,sep=""))
+    points(coef(modele3)[i], coef(modele3)[j],pch=3)
+    lines(c(IC3[1,i],IC3[1,i],IC3[2,i],IC3[2,i],IC3[1,i]),
+          c(IC3[1,j],IC3[2,j],IC3[2,j],IC3[1,j],IC3[1,j]),lty=2)
+  }}
```

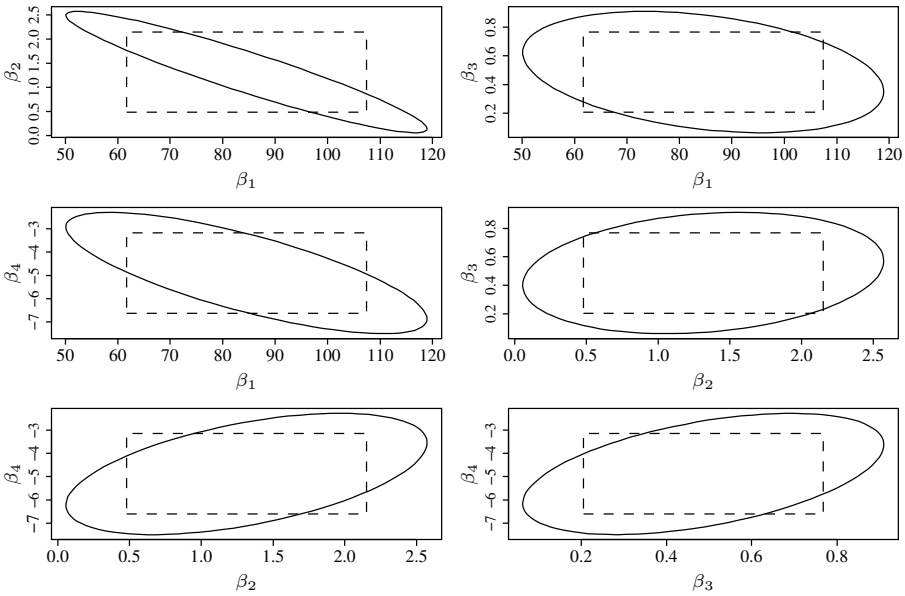


Fig. 5.2 – Régions de confiance et rectangle des couples de paramètres.

Afin d'observer la corrélation entre les paramètres, nous pouvons regarder l'orientation du grand axe de l'ellipse. Si cet axe n'est pas parallèle aux axes du repère, il y a corrélation. Ainsi nous observons que $\hat{\beta}_1$ et $\hat{\beta}_2$ sont fortement corrélés. Il en va de même avec ($\hat{\beta}_1$, $\hat{\beta}_3$) et ($\hat{\beta}_1$, $\hat{\beta}_4$). Enfin rappelons que nous pouvons calculer un IC à 95 % pour $\hat{\sigma}^2$ avec les commandes suivantes :

```
> c(resume3$sigma^2*modele3$df.res/qchisq(0.975,modele3$df.res),
+   resume3$sigma^2*modele3$df.res/qchisq(0.025,modele3$df.res))
[1] 133.6699 305.3706
```

5.4 Prévision

Soit $x'_{n+1} = (x_{n+1,1}, \cdots, x_{n+1,p})$ une nouvelle valeur et nous voulons prédire y_{n+1}. Le modèle indique que

$$y_{n+1} = x'_{n+1}\beta + \varepsilon_{n+1},$$

avec les ε_i i.i.d. et qui suivent une $\mathcal{N}(0, \sigma^2)$. A partir des n observations, nous avons estimé $\hat{\beta}$ et nous prévoyons y_{n+1} par

$$\hat{y}^p_{n+1} = x'_{n+1}\hat{\beta}.$$

L'espérance et la variance de l'erreur de prévision $\varepsilon^p_{n+1} = y_{n+1} - \hat{y}^p_{n+1}$ valent :

$$\begin{aligned}
\mathbb{E}(y_{n+1} - \hat{y}^p_{n+1}) &= 0 \\
\mathrm{V}(\hat{y}^p_{n+1} - y_{n+1}) &= \mathrm{V}(x'_{n+1}(\hat{\beta} - \beta) - \varepsilon_{n+1}) \\
&= x'_{n+1}\,\mathrm{V}(\hat{\beta} - \beta)x_{n+1} + \sigma^2 \\
&= \sigma^2\left[x'_{n+1}(X'X)^{-1}x_{n+1} + 1\right].
\end{aligned}$$

Nous obtenons la proposition suivante.

Proposition 5.4 (IC de prévision)
Un IC *de niveau* $(1-\alpha)$ *pour* y_{n+1} *est donné par*

$$\left[x'_{n+1}\hat{\beta} \pm t_{n-p}(1-\alpha/2)\hat{\sigma}\sqrt{x'_{n+1}(X'X)^{-1}x_{n+1} + 1}\right].$$

Preuve
$\hat{\beta}$ suit une loi normale et x_{n+1} est fixe donc $\hat{y}^p_{n+1}$ suit une loi normale. La valeur aléatoire y_{n+1} à prévoir suit une loi normale $\mathcal{N}(x'_{n+1}\beta, \sigma^2)$ et est indépendante des $y_1, \cdots, y_n$ par l'hypothèse $\mathcal{H}_3$.
Nous avons donc que y_{n+1} est indépendant de $\hat{y}^p_{n+1} = x'_{n+1}\hat{\beta}$ car $\hat{\beta}$ est une fonction linéaire des $y_1, \cdots, y_n$. L'erreur de prévision $y_{n+1} - \hat{y}^p_{n+1}$ suit donc une loi normale dont les moyenne et variance ont été calculées. Nous avons ainsi

$$N = \frac{\hat{y}^p_{n+1} - y_{n+1}}{\sigma\sqrt{x'_{n+1}(X'X)^{-1}x_{n+1} + 1}} \sim \mathcal{N}(0, 1).$$

Or σ est inconnue et estimée par $\hat{\sigma}$. Nous utilisons la définition d'un Student : si N suit une loi normale centrée réduite, si D suit un χ^2 à d ddl et si N et D sont indépendants, alors le rapport $N/\sqrt{D/d}$ suit un Student à d ddl.

La proposition 5.3 p. 93 indique que $D = \hat{\sigma}^2(n-p)/\sigma^2$ suit un χ^2 à $(n-p)$ degrés de liberté et que D est indépendant de $\hat{\beta}$. Or $\hat{\sigma}^2$ dépend uniquement des $y_1, \cdots, y_n$ et est donc indépendant de y_{n+1}. Il en va de même pour D. Le caractère aléatoire de N provient de $\hat{\beta}$ et de y_{n+1}, nous en déduisons que N et D sont indépendants d'où

$$\frac{N}{\sqrt{\frac{D}{d}}} = \frac{\hat{y}^p_{n+1} - y_{n+1}}{\hat{\sigma}\sqrt{x'_{n+1}(X'X)^{-1}x_{n+1} + 1}} \sim \mathcal{T}(n-p), \tag{5.5}$$

l'intervalle de confiance découle de ce résultat. □

5.5 Les tests d'hypothèses

5.5.1 Introduction

Reprenons l'exemple de la prévision des pics d'ozone. Nous avons modélisé les pics d'ozone par `T12`, `Vx` et `Ne12`. Il paraît raisonnable de se poser les questions suivantes :

(a) est-ce que la valeur de `O3` est influencée par `Vx` ?
(b) y a-t-il un effet nébulosité ?
(c) est-ce que la valeur de `O3` est influencée par `Vx` ou `T12` ?

Rappelons que le modèle utilisé est le suivant :

$$\texttt{O3} = \beta_1 + \beta_2\texttt{T12} + \beta_3\texttt{Vx} + \beta_4\texttt{Ne12} + \varepsilon.$$

Nous pouvons écrire les trois questions précédentes en termes de test d'hypothèse :
(a) correspond à $\mathrm{H}_0 : \beta_3 = 0$, contre $\mathrm{H}_1 : \beta_3 \neq 0$;
(b) correspond à $\mathrm{H}_0 : \beta_4 = 0$, contre $\mathrm{H}_1 : \beta_4 \neq 0$;
(c) correspond à $\mathrm{H}_0 : \beta_2 = \beta_3 = 0$, contre $\mathrm{H}_1 :\ \beta_2 \neq 0$ ou $\beta_3 \neq 0$.

Remarquons que tous ces cas reviennent à tester la nullité d'un ou plusieurs paramètres en même temps. Dans le cas c) on parle de nullité simultanée des coefficients. Cela veut dire que sous l'hypothèse H_0 certains coefficients sont nuls, donc les variables correspondant à ces coefficients ne sont pas utiles. Ce cas de figure correspond par définition à comparer deux modèles emboîtés l'un dans l'autre (l'un est un cas particulier de l'autre).
Le plan d'expérience privé de ces variables sera noté X_0 et les colonnes de X_0 engendreront un sous-espace noté $\Im(X_0)$. Afin d'alléger les notations, nous noterons $\Im(X_0) = \Im_0$ et $\Im(X) = \Im_X$. Le niveau des tests sera fixé de façon classique à α.

5.5.2 Test entre modèles emboîtés

Rappelons tout d'abord le modèle et les hypothèses utilisées :

$$Y = X\beta + \varepsilon \quad \text{où} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I),$$

cela veut dire que $\mathbb{E}(Y) \in \Im_X$ espace engendré par les colonnes de X.
Pour faciliter les notations, supposons que nous souhaitons tester la nullité simultanée des q derniers coefficients du modèle avec $q \leq p$. Le problème s'écrit alors de la façon suivante :

$$\mathrm{H}_0 : \quad \beta_{p-q+1} = \cdots = \beta_p = 0 \quad \text{contre} \quad \mathrm{H}_1 : \ \exists j \in \{p-q+1, \cdots, p\} : \beta_j \neq 0.$$

Que signifie $\mathrm{H}_0 : \ \beta_{p-q+1} = \cdots = \beta_p = 0$ en termes de modèle ? Si les q derniers coefficients sont nuls, le modèle devient

$$Y = X_0\beta_0 + \varepsilon \quad \text{où} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I),$$

où la matrice X_0 est composée des $p-q$ premières colonnes de X. Les colonnes de X_0 engendrent un espace noté $\Im_0$ de dimension $p_0 = p - q$. Ce sous-espace est bien évidemment inclus dans $\Im_X$. Sous l'hypothèse nulle H_0, l'espérance de Y appartiendra à ce sous-espace.
Une fois que les hypothèses du test sont fixées, il faut proposer une statistique de test. Nous allons voir une approche géométrique assez intuitive. Une approche plus analytique basée sur les tests de rapport de vraisemblance maximum est à faire en exercice (voir exercice 5.7).

Approche géométrique

Considérons le sous-espace noté $\Im_0$. Nous avons écrit que sous $\mathrm{H}_0 : \mathbb{E}(Y) \in \Im_0$. Dans ce cas, la méthode des moindres carrés consiste à projeter Y non plus sur $\Im_X$ (et obtenir $\hat{Y}$) mais sur $\Im_0$ et obtenir $\hat{Y}_0$. Visualisons ces différentes projections sur le graphique suivant :

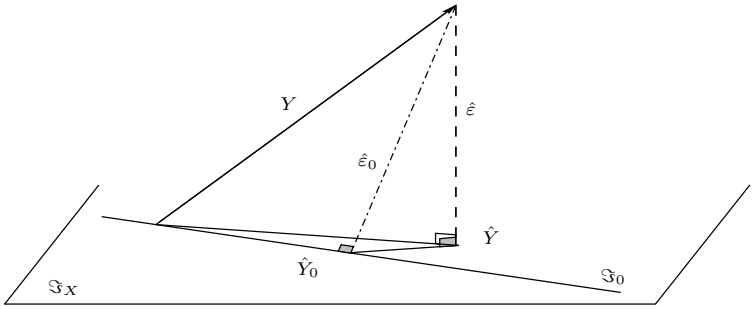


Fig. 5.3 – Représentation des projections.

L'idée intuitive du test, et donc du choix de conserver ou de rejeter H_0, est la suivante : si la projection de Y dans $\Im_0$, notée $\hat{Y}_0$, est « proche » de la projection de Y dans $\Im_X$, notée $\hat{Y}$, alors il semble logique de conserver l'hypothèse nulle. En effet, si l'information apportée par les deux modèles est la « même », il vaut mieux conserver le modèle le plus petit (principe de parcimonie). Il faut évidemment quantifier le terme « proche ». De manière naturelle, nous pouvons utiliser

la distance euclidienne entre $\hat{Y}_0$ et $\hat{Y}$, ou son carré, $\|\hat{Y}_0 - \hat{Y}\|^2$. Cependant, cette distance sera variable selon les données et selon les unités de mesures utilisées. Pour s'affranchir de ce problème d'échelle nous allons « standardiser » cette distance en la divisant par la norme au carré de l'erreur $\hat{\varepsilon}$. Les quantités $\hat{\varepsilon}$ et $\hat{Y}_0 - \hat{Y}$ n'appartiennent pas à des espaces de même dimension, nous divisons donc chaque terme par son degré de liberté respectif. Nous avons donc la statistique de test suivante :

$$F = \frac{\|\hat{Y}_0 - \hat{Y}\|^2/q}{\|Y - \hat{Y}\|^2/(n-p)} = \frac{\|\hat{Y}_0 - \hat{Y}\|^2/(p-p_0)}{\|Y - \hat{Y}\|^2/(n-p)}.$$

Pour utiliser cette statistique de test, il faut connaître sa loi au moins sous H_0. Remarquons que cette statistique est le rapport de deux normes au carré. Nous allons donc déterminer la loi du numérateur, du dénominateur et constater leur indépendance. Nous savons que

$$\hat{Y}_0 - \hat{Y} = P_{\Im_0} Y - P_{\Im_X} Y,$$

or $\Im_0 \subset \Im_X$ donc

$$\hat{Y}_0 - \hat{Y} = P_{\Im_0} P_{\Im_X} Y - P_{\Im_X} Y = (P_{\Im_0} - I_n) P_{\Im_X} Y = -P_{\Im_0^\perp} P_{\Im_X} Y.$$

Nous en déduisons que $(\hat{Y}_0 - \hat{Y}) \in \Im_0^\perp \cap \Im_X$ et donc que $(\hat{Y}_0 - \hat{Y}) \perp (Y - \hat{Y})$. La figure (5.3) permet de visualiser ces notions d'orthogonalité. Les vecteurs aléatoires $\hat{Y}_0 - \hat{Y}$ et $Y - \hat{Y}$ sont éléments d'espaces orthogonaux, ils ont donc une covariance nulle. Ces deux vecteurs sont des vecteurs gaussiens, ils sont donc indépendants et toute fonction fixe de ceux-ci reste indépendante, en particulier les normes du numérateur et du dénominateur sont indépendantes.
En utilisant l'hypothèse $\mathcal{H}_3$ de normalité et en appliquant le théorème de Cochran géométrique (théorème A.1 p. 418), nous en déduisons que ces deux normes suivent des lois du χ^2

$$\frac{1}{\sigma^2}\|P_{\Im_X^\perp} Y\|^2 \sim \chi^2_{n-p},$$
$$\frac{1}{\sigma^2}\|P_{\Im_0^\perp \cap \Im_X} Y\|^2 \sim \chi^2_{p-p_0}\left(\frac{1}{\sigma^2}\|P_{\Im_0^\perp \cap \Im_X} X\beta\|^2\right),$$

où le paramètre de décentrage $\|P_{\Im_0^\perp \cap \Im_X} X\beta\|^2/\sigma^2$ est nul sous H_0 puisque dans ce cas $X\beta \in \Im_0$. Nous pouvons conclure avec le théorème suivant.

Théorème 5.2 (Test entre modèles emboîtés)
Soit un modèle de régression à p variables $Y = X\beta + \varepsilon$ satisfaisant $\mathcal{H}_1$ et $\mathcal{H}_3$. Nous souhaitons tester la validité d'un sous-modèle (ou modèle emboîté) où un ou plusieurs coefficients sont nuls. Le plan d'expérience privé de ces variables sera noté X_0, les p_0 colonnes de X_0 engendreront un sous-espace noté $\Im_0$ et le sous-modèle sera $Y = X_0\beta_0 + \varepsilon$. Notons l'hypothèse nulle (modèle restreint) $\mathrm{H}_0 : \mathbb{E}(Y) \in \Im_0$ et l'hypothèse alternative (modèle complet) $\mathrm{H}_1 : \mathbb{E}(Y) \in \Im(X)$.

Pour tester ces deux hypothèses, nous utilisons la statistique de test F ci-dessous qui possède comme loi sous H_0 *:*

$$F = \frac{\|\hat{Y}_0 - \hat{Y}\|^2/(p-p_0)}{\|Y - \hat{Y}\|^2/(n-p)} \sim \mathcal{F}_{p-p_0,n-p},$$

et sous H_1 *la loi reste une loi de Fisher mais décentrée de* $\|P_{\Im_0^\perp \cap \Im_X} X\beta\|^2/\sigma^2$. *Notons aussi une écriture équivalente souvent utilisée et donc importante*

$$F = \frac{n-p}{p-p_0}\frac{\mathrm{SCR}_0 - \mathrm{SCR}}{\mathrm{SCR}} \sim \mathcal{F}_{p-p_0,n-p}.$$

L'hypothèse H_0 *sera repoussée en faveur de* H_1 *si l'observation de la statistique F est supérieure à* $f_{p-p_0,n-p}(1-\alpha)$, *la valeur α est le niveau du test.*

Preuve
La démonstration de la statistique de test F découle de la construction qui précède le théorème. En se rappelant que si $N \sim \chi^2$ à n ddl et $D \sim \chi^2$ à p ddl et si N et D sont indépendants alors

$$\frac{N}{D}\frac{d}{n} \sim \mathcal{F}_{n,p}.$$

L'écriture avec les SCR s'obtient en notant que

$$\begin{aligned}
\|Y - \hat{Y}_0\|^2 &= \|Y - P_{\Im_X} Y + P_{\Im_X} Y - P_{\Im_0} Y\|^2 \\
&= \|P_{\Im^\perp} Y + (I_n - P_{\Im_0}) P_{\Im_X} Y\|^2 \\
&= \|P_{\Im^\perp} Y\|^2 + \|P_{\Im_0^\perp \cap \Im_X} Y\|^2 \\
&= \|Y - \hat{Y}\|^2 + \|\hat{Y} - \hat{Y}_0\|^2.
\end{aligned}$$

Cette approche géométrique semble déconnectée des tests statistiques classiques mais il n'en est rien. Nous pouvons montrer (voir exercice 5.7) que le test F est tout simplement le test de rapport de vraisemblance maximale.

Test de Student de signification d'un coefficient β_j

Nous voulons tester $\mathrm{H}_0 : \beta_j = 0$ contre $\mathrm{H}_1 : \beta_j \neq 0$ (test bilatéral de significativité de β_j). Selon le théorème 5.2, la statistique de test est

$$F = \frac{\|\hat{Y} - \hat{Y}_0\|^2}{\hat{\sigma}^2}.$$

Nous rejetons H_0 si l'observation de la statistique F, notée $F(w)$, est telle que

$$F(\omega) \quad > \quad f_{1,n-p}(1-\alpha).$$

La statistique de test est un Fisher à 1 et $(n-p)$ ddl.

Ce test est équivalent (voir exercice 5.6) au test de « Student » à $(n-p)$ ddl qui permet de tester $\mathrm{H}_0 : \beta_j = 0$ contre $\mathrm{H}_1 : \beta_j \neq 0$ (test bilatéral de significativité de β_j) avec la statistique de test

$$T = \frac{\hat{\beta}_j}{\hat{\sigma}_{\hat{\beta}_j}}$$

qui suit sous H_0 une loi de Student à $(n-p)$ ddl. Nous rejetons H_0 si l'observation de la statistique T, notée $T(w)$, est telle que

$$|T(\omega)| \quad > \quad t_{n-p}(1-\alpha/2).$$

C'est sous cette forme que ce test figure dans les logiciels de régression linéaire.

Test de Fisher global

Si des connaissances *a priori* du phénomène étudié assurent l'existence d'un terme constant dans la régression, alors pour tester l'influence des régresseurs non constants sur la réponse, nous testerons l'appartenance de $\mathbb{E}(Y) = \mu$ à la diagonale $\Im_0(X) = \Delta$ de $\mathbb{R}^n$. Nous testerons ainsi la validité globale du modèle, c'est-à-dire que tous les coefficients sont supposés nuls, excepté la constante. Ce test est appelé test de Fisher global. Dans ce cas, $\hat{Y}_0 = \bar{Y}\mathbb{1}$ et nous avons la statistique de test suivante :

$$\frac{\|P_{\Im_X} Y - P_{\Im_0} Y\|^2/(p-1)}{\|Y - P_{\Im_X} Y\|^2/(n-p)} = \frac{\|P_{\Im_X} Y - \bar{Y}\mathbb{1}\|^2/(p-1)}{\|Y - P_{\Im_X} Y\|^2/(n-p)} \sim \mathcal{F}_{p-1,n-p}.$$

Si nous écrivons la statistique de test en utilisant le R^2, nous obtenons le rapport

$$F = \frac{\mathrm{R}^2}{1-\mathrm{R}^2}\frac{n-p}{p-1}.$$

Ce test est appelé par certains logiciels de statistique le test du R^2. La valeur de ce test est présenté par défaut dans le logiciel R sous le terme de `F-statistics`. Il faut faire attention à l'interprétation de ce test lorsque la constante n'est pas dans le modèle, voir exercice 5.4.

5.6 Applications

La concentration en ozone

Nous reprenons les données de l'ozone traitées précédemment dans ce chapitre et obtenons avec les commandes suivantes :

```
> modele3 <- lm(O3 ~ T12 + Vx + Ne12, data = ozone)
> resume3 <- summary(modele3)
> resume3
```

le tableau de résultats suivant :

```
Call:
lm(formula = O3 ~ T12 + Vx + Ne12, data = ozone)

Residuals:
    Min      1Q  Median      3Q     Max
-29.046  -8.482   0.786   7.702  28.292

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  84.5473    13.6067   6.214 1.38e-07 ***
T12           1.3150     0.4974   2.644  0.01117 *
Vx            0.4864     0.1675   2.903  0.00565 **
Ne12         -4.8934     1.0270  -4.765 1.93e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.91 on 46 degrees of freedom
Multiple R-Squared: 0.6819,     Adjusted R-squared: 0.6611
F-statistic: 32.87 on 3 and 46 DF,  p-value: 1.663e-11
```

La dernière ligne de la sortie du logiciel donne la statistique de test global, tous les coefficients sont nuls sauf la constante. Nous avons $n = 50$ observations, nous avons estimé 4 paramètres et donc le ddl du Fisher est bien $(3, 46)$. Nous refusons H_0 (tous les coefficients sauf la constante sont nuls) : au moins un des coefficients associé à `T12`, `Vx`, `Ne12` est non nul.

Le tableau `Coefficients` nous donne à la ligne j le test de la nullité d'un paramètre $\mathrm{H}_0 : \beta_j = 0$. Nous constatons qu'au seuil de 5 % tous les coefficients sont significativement différents de 0. La dernière colonne donne une version graphique du test : `***` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.001, `**` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.01, `*` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.05, `.` signifie que le test rejette H_0 pour des erreurs de première espèce supérieures ou égales à 0.1.

Tous les coefficients sont significativement non nuls et il ne semble donc pas utile de supprimer une variable explicative. Reprenons le modèle du chapitre précédent

```
> modele2 <- lm(O3 ~ T12 + Vx, data = ozone)
```

et comparons, à l'aide d'un test F, ces deux modèles emboîtés

```
> anova(modele2, modele3)
Model 1: O3 ~ T12 + Vx
Model 2: O3 ~ T12 + Vx + Ne12
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     47 13299.4
2     46  8904.6  1    4394.8 22.703 1.927e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nous retrouvons que le test F entre ces deux modèles est équivalent au test T de nullité du coefficient de la variable `Ne12` dans le modèle `modele3` (les deux probabilités critiques valent $1.93\ 10^{-5}$).
En conclusion, il semble que les 3 variables `T12, Vx` et `Ne12` soient explicatives de l'ozone.

La hauteur des eucalyptus

Le but de cet exemple est de prévoir la hauteur (`ht`) par la circonférence (`circ`). Lors des deux chapitres précédents nous avons introduit deux modèles de prévision, le modèle de régression simple

$$\texttt{ht} = \beta_1 + \beta_2 \texttt{circ} + \varepsilon$$

et le modèle de régression multiple

$$\texttt{ht} = \beta_1 + \beta_2 \texttt{circ} + \beta_3 \sqrt{\texttt{circ}} + \varepsilon.$$

Si l'on souhaite choisir le meilleur des deux modèles emboîtés, nous pouvons conduire un test F. Rappelons les commandes pour construire les deux modèles.

```
> regsimple <- lm(ht ~ circ, data = eucalypt)
> regM <- lm(ht ~ circ + I(sqrt(circ)), data = eucalypt)
```

Le test F est obtenu simplement comme suit.

```
> anova(regsimple, regM)
Analysis of Variance Table

Model 1: ht ~ circ
Model 2: ht ~ circ + I(sqrt(circ))
  Res.Df     RSS   Df Sum of Sq      F    Pr(>F)
1   1427 2052.08
2   1426 1840.66    1    211.43 163.80 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nous pouvons voir que l'observation de la statistique de test vaut 163.80, ce qui est supérieur au quantile 95 % d'une loi de Fisher à $(1, 1426)$ degré de liberté qui vaut 3.85 (obtenu avec `qf(0.95,1,regM$df.res)`).
Nous repoussons H_0 au profit de H_1 : le modèle de prévision adapté semble le modèle de régression multiple, malgré ses problèmes de prévision pour les hautes valeurs de circonférence. Rappelons que l'on peut retrouver le résultat de ce test avec le test T de nullité d'un coefficient :

```
> summary(regM)

Call:
lm(formula = ht ~ circ + I(sqrt(circ)), data = eucalypt)

Residuals:
     Min       1Q   Median       3Q      Max
-4.18811 -0.68811  0.04272  0.79272  3.74814

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   -24.35200    2.61444  -9.314   <2e-16 ***
circ           -0.48295    0.05793  -8.336   <2e-16 ***
I(sqrt(circ))   9.98689    0.78033  12.798   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.136 on 1426 degrees of freedom
Multiple R-Squared: 0.7922,     Adjusted R-squared: 0.7919
F-statistic:  2718 on 2 and 1426 DF,  p-value: < 2.2e-16
```

En effet, nous obtenons que l'observation de cette statistique vaut ici 12.798. Cette observation au carré est exactement égale à l'observation de la statistique de test F (en effet $12.798^2 \approx 163.80$). Par ailleurs les probabilités critiques sont bien égales. Notons que, dans ce résumé, le test de Fisher global repousse bien sûr l'hypothèse de nullité des coefficients des variables `circ` et $\sqrt{\texttt{circ}}$. L'observation de la statistique de test vaut ici 2718 alors que le quantile à 95 % d'une loi de Fisher à $(2, 1426)$ vaut 3.00. Cette réponse semblait évidente puisque repousser ici H_0 revient à dire qu'une des 2 variables au moins est explicative de la hauteur. Il est intéressant de noter que la variable `circ` intervient de manière négative (la hauteur décroît avec la circonférence) mais cela est compensé par la valeur positive de la variable $\sqrt{\texttt{circ}}$.
Nous pouvons aussi donner les intervalles de confiance pour le modèle et pour les prévisions. Pour cela, nous donnons une grille de valeurs de circonférences réparties entre le minimum (26 cm) et le maximum (77 cm), nous calculons la racine carrée de chaque élément de la grille et nous plaçons le tout dans un data-frame avec les mêmes noms que les variables du modèle.

```
> grille <- data.frame(circ = seq(min(eucalypt[,"circ"]),
+                                 max(eucalypt[,"circ"]), len = 100))
```

Ensuite nous utilisons la fonction `predict()` qui permet de donner les prévisions mais aussi les IC, tant pour le modèle que pour les prévisions. Enfin nous représentons les données et les IC à 95 %.

```
> ICdte <- predict(regM,new=grille,interval="conf",level=0.95)
> ICpre <- predict(regM,new=grille,interval="pred",level=0.95)
> plot(ht ~ circ, data = eucalypt, pch="+", col="grey60")
> matlines(grille,cbind(ICdte,ICpre[,-1]),lty=c(1,2,2,3,3),col=1)
> legend("topleft", lty=2:3, c("E(Y)","Y"))
```

Cette figure permet d'apprécier la mauvaise précision du modèle pour les fortes valeurs de circonférence.

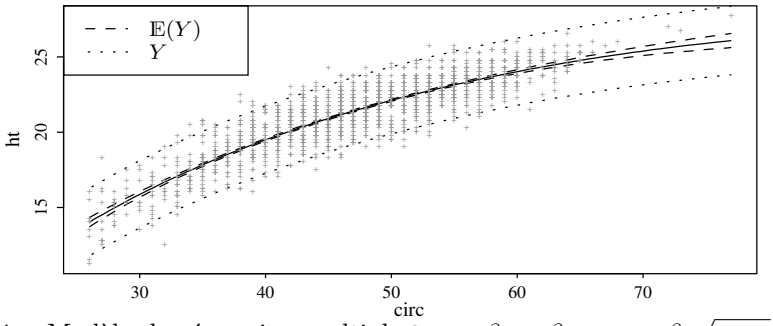


Fig. 5.4 – Modèle de régression multiple $\texttt{ht} = \beta_1 + \beta_2\texttt{circ} + \beta_3\sqrt{\texttt{circ}} + \varepsilon$ et intervalles de confiance à 95 % pour `ht` et pour $\mathbb{E}(\texttt{ht})$.

Nous aurions pu construire un modèle de prévision de la hauteur avec la racine carrée de la circonférence uniquement. Cependant, le test de ce modèle ($\texttt{ht} = \beta_1 + \beta_2\sqrt{\texttt{circ}}$) contre celui incorporant `circ` et $\sqrt{\texttt{circ}}$, ($\texttt{ht} = \beta_1 + \beta_2\texttt{circ} + \beta_3\sqrt{\texttt{circ}} + \varepsilon$), conduit à garder ce dernier.

5.7 Exercices

Exercice 5.1 (Questions de cours)

1) Nous pouvons justifier les MC quand $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$ *via* l'application du maximum de vraisemblance :
   A. oui,
   B. non,
   C. aucun rapport entre les deux méthodes.
2) Les estimateurs $\hat{\beta}$ des MC et $\widetilde{\beta}$ du maximum de vraisemblance sont-ils différents ?
   A. oui,
   B. non,
   C. pas toujours, cela dépend de la loi des erreurs.

3) Les estimateurs $\hat{\sigma}^2$ des MC et $\widetilde{\sigma}^2$ du maximum de vraisemblance sont-ils différents ?
A. oui,
B. non,
C. pas toujours, cela dépend de la loi des erreurs.

4) Le rectangle formé par les intervalles de confiance de niveau α individuels de β_1 et β_2 correspond à la région de confiance simultanée de niveau α de la paire (β_1, β_2) :
A. oui,
B. non,
C. cela dépend des données.

5) Nous avons n observations et p variables explicatives, nous supposons que ε suit une loi normale, nous voulons tester $\mathcal{H}_0 : \beta_2 = \beta_3 = \beta_4 = 0$. La loi de la statistique de test est :
A. $\mathcal{F}_{p-3,n-p}$,
B. $\mathcal{F}_{3,n-p}$,
C. une autre loi.

Exercice 5.2 (Théorème 5.1)
Démontrer le théorème 5.1 p. 94.

Exercice 5.3 (Test et $\mathbf{R^2}$)
Démontrer que la statistique du test Fisher F peut s'écrire sous la forme

$$F = \frac{\mathrm{R}^2 - \mathrm{R}_0^2}{1 - \mathrm{R}^2} \frac{n-p}{p-p_0},$$

où R^2 (R_0^2) correspond au R^2 du modèle complet (du modèle sous H_0).

Exercice 5.4 (Test et $\mathbf{R^2}$ et constante dans le modèle)
A FAIRE

Exercice 5.5 (Ozone)
Nous voulons expliquer la concentration de l'ozone sur Rennes en fonction des variables `T9`, `T12`, `Ne9`, `Ne12` et `Vx`. Les sorties données par R sont :

```
Coefficients :
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	62	10	$\boxed{1}$	0
T9	-4	$\boxed{2}$	-5	0
T12	5	0.75	$\boxed{3}$	0
Ne9	-1.5	1	$\boxed{4}$	0.08
Ne12	-0.5	0.5	$\boxed{5}$	0.53
Vx	0.8	0.15	5.5	0

```
--
Multiple R-Squared: 0.6666, Adjusted R-squared: 0.6081
Residual standard error: 16 on 124 degrees of freedom
F-statistic: 6 on 7 and 8 DF, p-value: 0
```

1) Compléter approximativement la sortie ci-dessus.
2) Rappeler la statistique de test et tester la nullité des paramètres séparément au seuil de 5 %.

3) Rappeler la statistique de test et tester la nullité simultanée des paramètres autres que la constante au seuil de 5 %.
4) Voici une nouvelle valeur, peut-on effectuer la prévision et donner un intervalle de confiance à 95 % (T9=10, T12=20, Ne9=0, Ne12=0, Vx=1) ?
5) Les variables Ne9 et Ne12 ne semblent pas influentes et nous souhaitons tester la nullité simultanée de $\beta_{\texttt{Ne9}}$ et $\beta_{\texttt{Ne12}}$. Proposer un test permettant de tester la nullité simultanée de $\beta_{\texttt{Ne9}}$ et $\beta_{\texttt{Ne12}}$ et l'effectuer à partir des résultats numériques suivants :

```
Coefficients :
                Estimate  Std. Error  t value  Pr(>|t|)
(Intercept)           66          11        6         0
T9                    -5           1       -5         0
T12                    6        0.75        8         0
Vx                     1         0.2        5         0
--
Multiple R-Squared: 0.5, Adjusted R-squared: 0.52
Residual standard error: 16.5 on 126 degrees of freedom
```

Exercice 5.6 (†Equivalence du test T et du test F)
Nous souhaitons tester la nullité d'un paramètre. Démontrer l'équivalence entre le test de Student et le test de Fisher.

Exercice 5.7 (††Equivalence du test F et du test de VM)
Nous souhaitons tester la nullité simultanée de q paramètres. Ecrire le test de rapport de vraisemblance maximale. Montrer que ce test est équivalent au test F.

Exercice 5.8 (††Test de Fisher pour une hypothèse linéaire quelconque)
Une hypothèse linéaire quelconque H_0 est de la forme $R\beta - r = 0$, où R est une matrice de taille $q \times p$ de rang q et r un vecteur de taille q.
Considérons un modèle de régression à p variables $Y = X\beta + \varepsilon$ satisfaisant $\mathcal{H}_1$ et $\mathcal{H}_3$. Nous souhaitons tester dans le cadre de ce modèle la validité d'une hypothèse linéaire quelconque $\mathrm{H}_0 \quad R\beta = r$, avec le rang de R égal à q, contre $\mathrm{H}_1 \quad R\beta \neq r$. Soit $\Im_0$ le sous-espace de $\Im_X$ de dimension $(p-q)$ engendré par la contrainte $R\beta = r$ (ou H_0) et $\Im_X$ le sous-espace de dimension p associé à H_1.
Démontrer que pour tester ces deux hypothèses nous utilisons la statistique de test F ci-dessous qui possède comme loi sous H_0 :

$$\begin{aligned} F &= \frac{\|\hat{Y} - \hat{Y}_0\|^2 / \dim(\Im_0^\perp \cap \Im_X)}{\|Y - \hat{Y}\|^2 / \dim(\Im_{X^\perp})} \\ &= \frac{n-p}{q} \frac{\|Y - \hat{Y}_0\|^2 - \|Y - \hat{Y}\|^2}{\|Y - \hat{Y}\|^2} \\ &= \frac{n-p}{q} \frac{\mathrm{SCR}_0 - \mathrm{SCR}}{\mathrm{SCR}} \sim \mathcal{F}_{q,n-p}. \end{aligned}$$

et sous H_1 la loi reste une loi de Fisher mais décentrée de $\|P_{\Im_0^\perp \cap \Im_X} X\beta\|^2/\sigma^2$.

Exercice 5.9 (Généralisation de la régression ridge)
Soit le problème de minimisation suivant (ridge généralisé) :

$$\hat{\beta}_{\mathrm{RG}}(\tau_1, \ldots, \tau_p) = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|Y - X\beta\|^2 - \sum_{j=1}^{p} \tau_j(\beta_j^2).$$

Montrez qu'à l'optimum, $\hat{\beta}_{\mathrm{RG}} = (X'X - \Delta)^{-1}X'Y$, où $\Delta = \mathrm{diag}(\dots, \delta_j, \dots)$. En déduire que le nombre effectif de paramètres est $\mathrm{tr}(X(X'X - \Delta)^{-1}X')$.

Exercice 5.10 (††IC pour la régression ridge)
Soit un modèle de régression $Y = X\beta + \varepsilon$ pour lequel nous nous intéressons à la régression ridge. Les variables sont supposées déjà centrées-réduites. Nous allons considérer que $\tilde{\kappa}$ est un coefficient fixé. Nous supposerons vérifiée l'hypothèse $\mathcal{H}_3$ de normalité des résidus. Nous nous plaçons dans le cas où la régression ridge est utile, c'est-à-dire $X\hat{\beta}_{\mathrm{ridge}}(\tilde{\kappa}) \neq P_X Y$.

1) Dans le cadre de la régression des MC pour $Y = X\beta + \varepsilon$, rappeler la loi de $\hat{\beta}$.
2) Rappeler la définition de l'estimateur $\hat{\beta}_{\mathrm{ridge}}(\tilde{\kappa})$.
3) Trouver la loi de $\hat{\beta}_{\mathrm{ridge}}(\tilde{\kappa})$.
4) Soit l'estimateur de σ^2 issu de la régression ridge : $\hat{\sigma}^2_{\mathrm{ridge}} = \|Y - \hat{Y}_{\mathrm{ridge}}\|^2/(n - \mathrm{tr}(H^*(\tilde{\kappa}))$, où $\mathrm{tr}(H^*(\tilde{\kappa}))$ est le nombre effectif de paramètres de la régression ridge. Montrer que le vecteur aléatoire $Y - \hat{Y}_{\mathrm{ridge}}$ n'est pas orthogonal à $\hat{Y}_{MC}$.
5) Trouver le point de la démonstration du théorème 5.3 qui n'est pas assuré avec l'estimateur $\hat{\beta}_{\mathrm{ridge}}$ et l'estimateur $\hat{\sigma}^2_{\mathrm{ridge}}$. Nous en déduisons alors qu'il n'est plus assuré que l'intervalle de confiance de β en régression ridge soit de la forme énoncée par le théorème 5.1 (en remplaçant $\hat{\beta}$ par $\hat{\beta}_{\mathrm{ridge}}$ et $\hat{\sigma}^2$ par $\hat{\sigma}^2_{\mathrm{ridge}}$).
6) Concevoir un algorithme calculant les IC par bootstrap pour chaque coordonnée de $\hat{\beta}_{\mathrm{ridge}}$, avec $\tilde{\kappa}$ considéré comme fixé.
7) Généraliser la question précédente en incluant la détermination de $\tilde{\kappa}$.

5.8 Notes

Quelquefois l'hypothèse de normalité ($\mathcal{H}_3$), nécessaire à la validité des tests et des intervalles de confiance, n'est pas vérifiée ou non vérifiable. Les tests qui permettent de choisir entre des modèles contraints ou des modèles non contraints (ou tests entre modèles emboîtés) peuvent être alors remplacés par une des procédures de choix de modèles décrites au chapitre 7.
Pour les intervalles de confiance, une procédure spécifique existe, basée sur le bootstrap. Nous présenterons également des résultats asymptotiques.

5.8.1 Intervalle de confiance : bootstrap

L'objectif de cette section est de présenter la méthode du bootstrap en régression afin que le lecteur puisse obtenir un intervalle de confiance pour β, sans donner d'hypothèse supplémentaire sur la loi des erreurs ε. Le lecteur intéressé par le bootstrap en tant que méthode statistique pourra consulter le livre de Efron & Tibshirani (1993).
Le modèle utilisé est $Y = X\beta + \varepsilon$ où ε est une variable aléatoire de loi F inconnue et d'espérance nulle. L'idée du bootstrap est d'estimer cette loi par ré-échantillonnage.
Nous considérons que la constante fait partie du modèle. La somme des résidus estimés vaut donc zéro.

— A partir du nuage de points (X, Y), estimer par les MC β et ε par $\hat{\beta}$ et $\hat{\varepsilon}$. Soit $\hat{F}_n$ la distribution empirique des $\hat{\varepsilon}$.
— Tirer au hasard avec remise n résidus estimés $\hat{\varepsilon}_i$ notés $\hat{\varepsilon}^*_i$.
— A partir de ces n résidus, construire un échantillon

$$Y^* = X\hat{\beta} + \hat{\varepsilon}^*$$

appelé échantillon bootstrapé ou encore échantillon étoile.

— A partir de l'échantillon étoile (X, Y^*) estimer le vecteur des paramètres. La solution est

$$\hat{\beta}^* = (X'X)^{-1}X'Y^*.$$

La théorie du bootstrap indique que la distribution de $\sqrt{n}(\hat{\beta}^* - \hat{\beta})$, distribution que nous pouvons calculer directement à partir des données, approche correctement la distribution de $\sqrt{n}(\hat{\beta} - \beta)$ qui elle ne peut pas être calculée, puisque β est inconnu.
Afin de calculer la distribution de $\sqrt{n}(\hat{\beta}^* - \hat{\beta})$ nous calculons B échantillons bootstrapés ou étoiles et calculons ensuite B estimateurs $\hat{\beta}^*$ de $\hat{\beta}$.
Il faut donc répéter B fois les étapes suivantes :

— tirer au hasard avec remise n résidus estimés $\hat{\varepsilon}_i$ notés $\hat{\varepsilon}_i^{(k)}$;
— à partir de ces n résidus, construire un échantillon $y_i^{(k)} = x_i\hat{\beta} + \hat{\varepsilon}_i^{(k)}$, appelé échantillon bootstrapé ;
— à partir de cet échantillon bootstrapé, estimer $\hat{\beta}^{(k)}$.

Pour donner un ordre d'idée, une valeur de 1000 pour B est couramment utilisée. Nous obtenons alors B estimateurs de β noté $\hat{\beta}^{(k)}$. A partir de ces 1000 valeurs, nous pouvons calculer toutes les statistiques classiques. Si nous nous intéressons à la distribution des $\hat{\beta}_j$, nous pouvons estimer cette distribution en calculant l'histogramme des $\hat{\beta}_j^{(k)}$. De même un intervalle de confiance peut être obtenu en calculant les quantiles empiriques des $\hat{\beta}_j^{(k)}$.

Voyons cela sur l'exemple de la concentration en ozone. Nous continuons notre modèle à 3 variables explicatives des pics d'ozone, la température à 12 h (`T12`), la nébulosité à 12 h (`Ne12`) et la projection du vent à 12 h sur l'axe est-ouest (`Vx`). Le modèle est toujours construit grâce à la commande suivante :

```
> modele3 <- lm(O3 ~ T12 + Vx + Ne12, data = ozone)
```

Nous pouvons résumer la phase d'estimation et nous intéresser aux coefficients.

```
> resume3 <- summary(modele3)
> resume3$coefficients[,1:2]
              Estimate Std. Error
(Intercept) 84.5473326 13.6067253
T12          1.3150459  0.4974102
Vx           0.4864456  0.1675496
Ne12        -4.8933729  1.0269960
```

Cette procédure ne suppose que deux hypothèses très faibles $\mathcal{H}_1$ et $\mathcal{H}_2$. Afin de construire un intervalle de confiance pour les paramètres sans supposer la normalité, nous appliquons la procédure de bootstrap.
La première étape consiste à calculer les résidus estimés $\hat{\varepsilon} = \hat{Y} - Y$ et ajustements $\hat{Y}$.

```
> res <- residuals(modele3)
> ychap <- predict(modele3)
> COEFF <- matrix(0, ncol = 4, nrow = 1000)
> colnames(COEFF) <- names(coef(modele3))
> ozone.boot <- ozone
```

Ensuite nous allons appliquer la procédure de bootstrap avec $B = 1000$ échantillons bootstrapés.

```
> for(i in 1:nrow(COEFF)){
+   resetoile <- sample(res, length(res), replace=T)
+   O3etoile <- ychap + resetoile
+   ozone.boot[,"O3"] <- O3etoile
+   regboot <- lm(formula(modele3), data=ozone.boot)
+   COEFF[i,] <- coef(regboot)
+ }
```

Nous avons obtenu une matrice de 1000 coefficients estimés (`COEFF`) et nous choisissons les quantiles empiriques à 2.5 % et 97.5 % de ces échantillons afin de déterminer les intervalles de confiance.

```
> apply(COEFF, 2, quantile, probs = c(0.025,0.975))

      (Intercept)       T12        Vx      Ne12
2.5%     57.88882 0.3815498 0.1689669 -6.803680
97.5%   109.99762 2.3447899 0.7837472 -2.972169
```

Un IC à 95 % pour le coefficient associé à `T12` est donc donné par $[0.41; 2.26]$. En supposant que les erreurs suivent une loi normale, nous avions $[0.48; 2.15]$. L'intervalle est donc plus grand.
Nous pouvons aussi considérer un estimateur de la densité des $\hat{\beta}_j$ en traçant un histogramme des $\hat{\beta}_j^*$. Voici l'histogramme des estimateurs du coefficient associé à la variable température :

```
> hist(COEFF[,"T12"], main = "", xlab = "Coefficient de T12")
```

Cet histogramme semble indiquer que la loi est proche d'une loi normale.

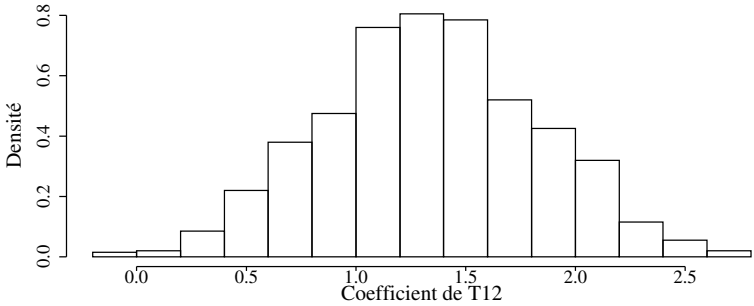


Fig. 5.5 – Histogramme des estimateurs bootstrapés pour la variable `T12`.

Nous aurions pu commencer par tirer avec remise n individus parmi les n couples d'observations (x_i', y_i) et continuer comme présenté ci-dessus. Ce bootstrap est plus adapté au cas où les variables X_j sont des variables aléatoires. Les lecteurs intéressés par cette procédure peuvent consulter Efron & Morris (1973) par exemple.

5.8.2 Test de Fisher pour une hypothèse linéaire quelconque

Dans la partie 5.5.2, nous avons testé la nullité simultanée d'un certain nombre de coefficients. Cela nous a permis de transcrire facilement l'hypothèse H_0 en termes d'espace. Nous allons aborder maintenant le cas où l'hypothèse à tester est de la forme $R\beta = r$.

Rappelons nos trois questions initiales :

(a) est-ce que la valeur de `O3` est influencée par `Vx` ?
(b) y a-t-il un effet nébulosité ?
(c) est-ce que la valeur de `O3` est influencée par `Vx` et `T12` ?

Toutes ces hypothèses sont des cas particuliers de l'hypothèse linéaire générale :

$$\mathrm{H}_0 : \ R\beta = 0 \text{ contre } \quad \mathrm{H}_1 : \ R\beta \neq 0,$$

où R est une matrice $q \times p$ connue de rang q. Il suffit de poser

$$\begin{array}{lllll}
(a) & R & = & \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} & r = 0 \ ; \\
(b) & R & = & \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} & r = 0 \ ; \\
(c) & R & = & \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} & r = \begin{bmatrix} 0 \\ 0 \end{bmatrix},
\end{array}$$

où R est une matrice $q \times p$ de rang q, r est un vecteur de taille q et où R et r sont connus. Nous imposons q contraintes linéaires **2 à 2 indépendantes** sur les coefficients β_j. Cette façon de procéder permet de tester également

$$\begin{array}{llllll}
\beta_j = \beta_j^0 & R & = & \begin{bmatrix} 0 & \cdots & 1_j & \cdots & 0 \end{bmatrix} & r = \beta_j^0 \ ; \\
\beta_j = \beta_i & R & = & \begin{bmatrix} 0 & \cdots & 1_i & \cdots & -1_j & \cdots & 0 \end{bmatrix} & r = 0 \ ;
\end{array}$$

où encore les q derniers β_j sont nuls grâce aux matrices

$$R_{q\times p} \quad = \quad \begin{bmatrix} 0 & I_q \end{bmatrix} \qquad r = 0_q.$$

Nous imposons la contrainte générale $R\beta = r$. Cela revient à imposer en fait q (le rang de R) contraintes linéaires sur les paramètres et cela peut se traduire d'un point de vue géométrique par $\mathbb{E}(Y)$ n'appartient plus à l'espace engendré par toutes les colonnes de X, espace que nous avons noté $\Im(X)$ mais à un sous-espace engendré par les colonnes de X satisfaisant la contrainte linéaire $R\beta = 0$.

Définition 5.1
Une hypothèse linéaire quelconque H_0 *est de la forme* $R\beta - r = 0$*, où* R *est une matrice de taille* $q \times p$ *de rang* q *et* r *un vecteur de taille* q*.*

Théorème 5.3
Soit un modèle de régression à p *variables* $Y = X\beta + \varepsilon$ *satisfaisant* $\mathcal{H}_1$ *et* $\mathcal{H}_3$*. Nous souhaitons tester dans le cadre de ce modèle la validité d'une hypothèse linéaire quelconque* H_0 $R\beta = r$*, avec le rang de* R *égal à* q*, contre* H_1 $R\beta \neq r$*. Soit* $\Im_0$ *le sous-espace de* $\Im_X$ *de dimension* $(p-q)$ *engendré par la contrainte* $R\beta = r$ *(ou* H_0*) et* $\Im_X$ *le sous-espace de dimension* p *associé à* H_1*.*

Pour tester ces deux hypothèses nous utilisons la statistique de test F ci-dessous qui possède comme loi sous H_0 *:*

$$\begin{aligned} F &= \frac{\|\hat{Y}-\hat{Y}_0\|^2/\dim(\Im_0^\perp \cap \Im_X)}{\|Y-\hat{Y}\|^2/\dim(\Im_{X^\perp})} \\ &= \frac{n-p}{q}\frac{\|Y-\hat{Y}_0\|^2-\|Y-\hat{Y}\|^2}{\|Y-\hat{Y}\|^2} \\ &= \frac{n-p}{q}\frac{\mathrm{SCR}_0-\mathrm{SCR}}{\mathrm{SCR}} \sim \mathcal{F}_{q,n-p}. \end{aligned}$$

et sous H_1 *la loi reste une loi de Fisher mais décentrée de* $\|P_{\Im_0^\perp \cap \Im_X} X\beta\|^2/\sigma^2$*. L'estimation* $\hat{Y}_0$ *est donnée par*

$$X\hat{\beta}_0 = X\hat{\beta} + X(X'X)^{-1}R'[R(X'X)^{-1}R']^{-1}(r-R\hat{\beta}).$$

L'hypothèse H_0 *sera repoussée en faveur de* H_1 *si l'observation de la statistique F est supérieure à* $f_{q,n-p}(1-\alpha)$*, la valeur* α *étant la probabilité de rejeter à tort* H_0 *ou erreur de première espèce.*

Preuve
Nous pouvons toujours traduire l'hypothèse H_0 : $R\beta = r$ en termes de sous-espace de $\Im_X$. Lorsque $r = 0$, nous avons un sous-espace vectoriel de $\Im_X$ et lorsque $r \neq 0$ nous avons un sous-espace affine de $\Im_X$. Dans les deux cas, nous noterons ce sous-espace $\Im_0$ et $\Im_0 \subset \Im_X$. Cependant nous ne pourrons plus le visualiser facilement comme nous l'avons fait précédemment avec $\Im_{X_0}$ où nous avions enlevé des colonnes à la matrice X. Nous allons décomposer l'espace $\Im_X$ en deux sous-espaces orthogonaux

$$\Im_X = \Im_0 \overset{\perp}{\oplus} (\Im_0^\perp \cap \Im_X).$$

Sous H_0, l'estimation des moindres carrés donne $\hat{Y}_0$ projection orthogonale de Y sur $\Im_0$ et nous appliquons la même démarche pour construire la statistique de test. La démonstration est donc la même que celle du théorème 5.2. C'est-à-dire que nous regardons si $\hat{Y}_0$ est proche de $\hat{Y}$ et nous avons donc

$$\begin{aligned} F &= \frac{\|\hat{Y}-\hat{Y}_0\|^2/\dim(\Im_0^\perp \cap \Im_X)}{\|Y-\hat{Y}\|^2/\dim(\Im_{X^\perp})} \\ &= \frac{n-p}{q}\frac{\|Y-\hat{Y}_0\|^2-\|Y-\hat{Y}\|^2}{\|Y-\hat{Y}\|^2} \\ &= \frac{n-p}{q}\frac{\mathrm{SCR}_0-\mathrm{SCR}}{\mathrm{SCR}} \sim \mathcal{F}_{q,n-p}. \end{aligned}$$

Le problème du test réside dans le calcul de $\hat{Y}_0$. Dans la partie précédente, il était facile de calculer $\hat{Y}_0$ car nous avions la forme explicite du projecteur sur $\Im_0$.
Une première façon de procéder revient à trouver la forme du projecteur sur $\Im_0$. Une autre façon de faire est de récrire le problème de minimisation sous la contrainte $R\beta = r$. Ces deux manières d'opérer sont présentées en détail dans la correction de l'exercice 2.13. Dans tous les cas l'estimateur des MC contraints par $R\beta = r$ est défini par

$$\hat{\beta}_0 = \hat{\beta} + (X'X)^{-1}R'[R(X'X)^{-1}R']^{-1}(r-R\hat{\beta}).$$

5.8.3 Propriétés asymptotiques

Nous allons étudier des propriétés asymptotiques de $\hat{\beta}$ et $\hat{\sigma}^2$ lorsque la taille de l'échantillon n tend vers l'infini. Ce type d'études permet de s'assurer de la pertinence des estimateurs. En effet, nous savons que pour n fixé, l'estimateur $\hat{\beta}$ est sans biais pour β, et de variance minimum parmi les estimateurs linéaires sans biais (théorème de Gauss-Markov). Mais lorsque n augmente, c'est-à-dire lorsque la quantité de données et donc l'information augmente, est-ce que $\hat{\beta}$ va tendre vers β ? Sous quelles conditions cette convergence (en moyenne quadratique) se produit-elle ?
Commençons par introduire quelques notations. Pour chaque n, les données sont notées par $(y_n, X_n\beta, \sigma^2 I_n)$. Nous supposerons que la matrice X_n, de dimension $n \times p$, est de rang p pour tout $n > m$. Pour $n > m$, nous définissons la suite $\hat{\beta}_n$ par

$$\hat{\beta}_n = (X_n' X_n)^{-1} X_n' y_n,$$

avec $\mathbb{E}(\hat{\beta}_n) = \beta$ et $\mathrm{V}(\hat{\beta}_n) = \sigma^2 (X_n' X_n)^{-1}$.

Une condition suffisante pour que la suite $(\hat{\beta}_n)$ converge vers β en moyenne quadratique, et donc en probabilité, est que $\mathrm{V}(\hat{\beta}_n)$ converge vers zéro.

Théorème 5.4 (Convergence de $\hat{\beta}$)
Sous $\mathcal{H}_1$ et $\mathcal{H}_2$, si $(X_n' X_n)^{-1}$ tend vers zéro avec n alors $\hat{\beta}_n$ converge vers β en moyenne quadratique et en probabilité.

Rappelons que si les variables $X_1, \ldots, X_p$ sont supposées aléatoires, ce qui n'est pas le cas ici, $(X_n' X_n)/n$ est un estimateur de la matrice de variance des p variables explicatives. *A priori*, si les X_j sont mesurées, nous pouvons supposer qu'elles sont mesurées avec des erreurs, même petites. Cela permet de penser que ces variables peuvent être considérées comme aléatoires. En pratique, nous supposons toujours qu'un vecteur aléatoire admet une matrice de variance A fixée et donc *a priori* $(X_n' X_n)/n \to A$. Comme A est fixée, nous avons $n(X_n' X_n)^{-1} \to A^{-1}$ et donc $(X_n' X_n)^{-1} \to 0$. La condition de convergence n'est donc absolument pas contraignante.
Nous savons que $\mathrm{V}(\hat{\beta}_n) = \sigma^2 (X_n' X_n)^{-1}$, où σ^2 est fixé. La condition de convergence s'exprime donc comme « la variabilité de $\hat{\beta}_n$ tend vers 0 avec n », ce qui semble assez naturel.

Preuve

$$\begin{aligned}
\mathbb{E}\left(\|\hat{\beta}_n - \beta\|^2\right) &= \mathbb{E}\left(\|\hat{\beta}_n - \mathbb{E}\hat{\beta}_n\|^2\right) \\
&= \mathrm{tr}[\mathrm{V}(\hat{\beta}_n)] \\
&= \sigma^2 tr (X_n' X_n)^{-1}.
\end{aligned}$$

Si $(X_n' X_n)^{-1}$ tend vers zéro, $\mathrm{tr}(X_n' X_n)^{-1})$ tend vers zéro et le théorème est démontré.□

Nous pouvons aussi considérer l'estimateur $\hat{\sigma}^2$ et se poser la même question : quelles sont les conditions nécessaires pour que l'estimateur converge vers sa vraie valeur quand le nombre de données augmente ?

Théorème 5.5 (Convergence de $\hat{\sigma}^2$)
Si $\mathcal{H}_1$ et $\mathcal{H}_2$ sont vérifiées et si les ε_i sont i.i.d. avec $\mathbb{E}[\varepsilon^2] = \sigma^2 < \infty$, alors $\hat{\sigma}_n^2$ converge vers σ^2 en probabilité.

Preuve
Partons de la définition de $\hat{\sigma}_n^2$:

$$\begin{aligned}
\hat{\sigma}_n^2 &= \frac{1}{n-p}\hat{\varepsilon}_n'\hat{\varepsilon}_n \\
&= \frac{1}{n-p}\varepsilon_n'(I - P_{X_n})\varepsilon_n \\
&= \frac{1}{n-p}\varepsilon_n'\varepsilon_n - \frac{1}{n-p}\varepsilon_n' P_{X_n}\varepsilon_n.
\end{aligned}$$

La loi forte des grands nombres indique que le premier terme converge p.s. vers σ^2. Nous allons montrer que le second terme converge en probabilité vers zéro. Pour tout η positif, l'inégalité de Markov donne

$$\begin{aligned}
\forall \eta > 0, \quad P\left(\frac{\varepsilon_n' P_{X_n}\varepsilon_n}{n-p} > \eta\right) &\leq \frac{1}{\eta(n-p)}\mathbb{E}(\varepsilon_n' P_{X_n}\varepsilon_n) \\
&\leq \frac{1}{\eta(n-p)}\mathbb{E}[\mathrm{tr}(\varepsilon_n' P_{X_n}\varepsilon_n)] \\
&\leq \frac{1}{\eta(n-p)}\sigma^2\,\mathrm{tr}(P_{X_n}) \\
&\leq \frac{p}{\eta(n-p)}\sigma^2 \to 0.
\end{aligned}$$

La dernière partie de cette note concerne la normalité asymptotique. Cela va permettre de donner des intervalles de confiance (IC) et de faire des tests sans supposer d'hypothèse supplémentaire sur la loi des ε car nous utiliserons alors la loi limite.

Théorème 5.6 (Normalité asymptotique)
Si $\mathcal{H}_1$ et $\mathcal{H}_2$ sont vérifiées, si les ε_i sont i.i.d. avec $\mathbb{E}[\varepsilon^2] = \sigma^2 < \infty$, et si $(X_n'X_n)/n$ tend vers A (strictement) définie positive (donc inversible), alors

$$\sqrt{n}(\hat{\beta}_n - \beta) \to \mathcal{N}(0, \sigma^2 A^{-1})$$

Preuve
Nous allons donner une idée de la preuve de ce théorème.

$$\begin{aligned}
\sqrt{n}(\hat{\beta}_n - \beta) &= \sqrt{n}\left((X_n'X_n)^{-1}X_n'y_n - \beta\right) \\
&= \sqrt{n}\left((X_n'X_n)^{-1}X_n'(X_n\beta + \varepsilon_n) - \beta\right) \\
&= \sqrt{n}\left((X_n'X_n)^{-1}X_n'\varepsilon_n\right) \\
&= \frac{1}{\sqrt{n}}\left(\left(\frac{X_n'X_n}{n}\right)^{-1} X_n'\varepsilon_n\right).
\end{aligned}$$

Posons w_i la i^{e} colonne de la matrice $n(X_n'X_n)^{-1}X_n'$. Nous utilisons ensuite un théorème central limite pondéré qui se trouve dans Antoniadis *et al.* (1992). □

Nous retrouvons la même condition sur la convergence de $(X_n'X_n)^{-1}$. Il semble raisonnable de penser que cette condition est vérifiée. Afin d'utiliser ce théorème en pratique, pour calculer des intervalles de confiance, nous devons savoir si, avec les observations et leur nombre, nous avons $(X_n'X_n)^{-1}$ suffisamment proche de 0. Le problème est bien sûr impossible à résoudre formellement et, comme cette convergence dépend à la fois de la taille de l'échantillon n et des observations X_n, il n'existe pas de règle du type « à partir de 100 observations on peut... ».

Hypothèse gaussienne

Nous supposons dorénavant que les résidus suivent une loi normale de moyenne nulle et de variance $\sigma^2\Omega$. Nous avons alors les propriétés classiques suivantes (dont la démonstration consiste à se ramener au modèle (*) et à faire comme pour les MC).

Proposition 5.5
i) $\hat{\beta}_{MCG}$ *est un vecteur gaussien de moyenne* β *et de variance* $\sigma^2(X'\Omega^{-1}X)^{-1}$.
ii) $\hat{\sigma}^2_{MCG}$ *vérifie* $(n-p)\hat{\sigma}^2_{MCG}/\sigma^2 \sim \chi^2_{n-p}$.
iii) $\hat{\beta}_{MCG}$ *et* $\hat{\sigma}^2_{MCG}$ *sont indépendants.*

Nous pouvons aussi tester une hypothèse linéaire quelconque.

Théorème 5.7
Soit un modèle de régression à p *variables* $Y = X\beta + \varepsilon$ *satisfaisant* $\mathcal{H}_1$ *et* $\mathcal{H}_3$. *Nous souhaitons tester dans le cadre de ce modèle la validité d'une hypothèse linéaire quelconque* $\mathrm{H}_0 : \quad R\beta = r$, *avec le rang de* R *égal à* q, *contre* $\mathrm{H}_1 : \quad R\beta \neq r$. *Soit* $\Im_0$ *le sous-espace de* $\Im_X$ *de dimension* $(p-q)$ *engendré par la contrainte* $R\beta = r$ *(ou* H_0*) et* $\Im_X$ *le sous-espace de dimension* p *associé à* H_1.
Pour tester ces deux hypothèses nous utilisons la statistique de test F *:*

$$F = \frac{\|r - R\hat{\beta}_{MCG}\|^2_{[R(X'\Omega^{-1}X)^{-1}R']^{-1}}}{\|Y - X\hat{\beta}_{MCG}\|^2_{\Omega^{-1}}} \frac{n-p}{q},$$

qui sous H_0 *suit la loi* $\mathcal{F}_{q,n-p}$. *L'hypothèse* H_0 *sera repoussée en faveur de* H_1, *au niveau* α *du test, si l'observation de la statistique* F *est supérieure à* $f_{q,n-p}(1-\alpha)$.

Les applications sont identiques à celles rencontrées en régression ordinaire et l'on peut citer par exemple les tests de Student de nullité d'un coefficient ou les tests de Fisher de nullité simultanée de q coefficients.

Chapitre 6

Variables qualitatives : ANCOVA et ANOVA

6.1 Introduction

Jusqu'à présent, les variables explicatives étaient quantitatives continues. Or il arrive fréquemment que certaines variables explicatives soient des variables qualitatives. Dans ce cas, pouvons-nous appliquer la méthode des moindres carrés que nous venons de voir ?
Reprenons l'exemple des eucalyptus, nous avons mesuré 1429 couples circonférence-hauteur. Parmi ces 1429 arbres, 527 proviennent d'une partie du champ notée bloc $A1$, 586 proviennent d'une autre partie du champ notée bloc $A2$ et 316 proviennent de la dernière partie du champ notée bloc $A3$. Le tableau suivant donne les 2 premières mesures effectuées dans chaque bloc :

Individu	`ht`	`circ`	`bloc`
1	18.25	36	$A1$
2	19.75	42	$A1$
528	17.00	38	$A2$
529	18.50	46	$A2$
1114	17.75	36	$A3$
1115	19.50	45	$A3$

Tableau 6.1 – Mesures pour 6 eucalyptus de la hauteur et la circonférence et du bloc (`ht`, `circ` et `bloc`).

Nous avons dorénavant 2 variables explicatives : la circonférence et la provenance de l'arbre. Pouvons-nous effectuer une régression multiple ? Comment utiliser la variable `bloc` ? Dans cet exemple simple, nous pouvons représenter les données avec en abscisse la circonférence, en ordonnée la hauteur et en couleur (par exemple) la provenance (fig. 6.1).

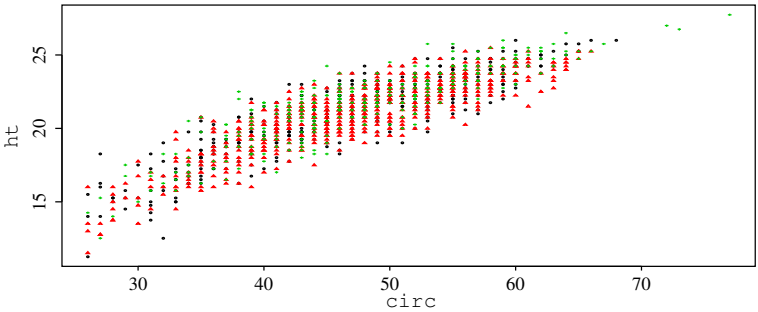


Fig. 6.1 – Nuage de points et régression simple pour chaque niveau de bloc. La provenance est représentée par un symbole (rond, triangle, +) différent.

La provenance pourrait avoir un effet sur la hauteur mais cela est difficile à observer. Afin d'intégrer la variable `bloc`, il faut commencer par la recoder car les calculs ne peuvent pas être effectués avec la variable en l'état. Chaque modalité est transformée en un vecteur d'indicatrices d'appartenance à la modalité :

$$\texttt{bloc} = A = \begin{bmatrix} A1 \\ A1 \\ A2 \\ A2 \\ A3 \\ A3 \end{bmatrix} \quad \Longrightarrow \quad A_c = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Ce codage, appelé codage disjonctif complet, remplace donc une variable admettant I modalités en I variables binaires[1]. Nous pouvons déjà remarquer que la somme des vecteurs colonnes de cette matrice A_c est égale au vecteur $\mathbb{1}$. En effet un individu i admet obligatoirement une modalité et une seule et possède donc toujours un unique 1 sur la i^{e} ligne de A_c.
Ce chapitre va traiter en détail l'analyse de la covariance[2], une variable Y est expliquée par une (ou des) variable(s) continue(s) et une (ou des) variable(s) qualitative(s). Puis nous présenterons rapidement l'analyse de la variance à un facteur (une variable Y est expliquée par une variable qualitative) et l'analyse de la variance à deux facteurs (deux variables qualitatives). Avant de présenter ces méthodes en détail, il nous semble important de rappeler comment écrire un modèle dans R. Considérons un data-frame appelé `donnees` où la variable à expliquer est `Y` et les variables explicatives sont `X1`,...,`X10`. Un modèle dans R s'écrit à l'aide d'une formule qui fait intervenir le symbole ∼

```
Y ~ X1, data = donnees
```

Les formules peuvent être utilisées dans des fonctions qui admettent des modèles telles que **plot**, **lm**... Nous allons présenter les principales écritures de formules

1. Variables binaires appelées *dummy variables* en anglais, c'est-à-dire variables fictives.
2. Nous noterons aussi cette analyse par l'acronyme anglo-saxon ANCOVA.

dans le cadre du modèle linéaire (fonction **lm**). Nous souhaitons tout d'abord estimer les paramètres du modèle

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon.$$

Nous écrivons alors :

```
lm(Y ~ X1 + X2, data = donnees)
```

La constante est automatiquement mise dans le modèle. Si nous ne souhaitons pas l'intégrer, nous écrivons :

```
lm(Y ~ X1 + X2 - 1, data = donnees)
```

Il faut faire attention au test de Fisher global présenté à la page 102 dans ce cas. Si nous souhaitons travailler avec toutes les variables sauf `X9` et `X10`, nous avons l'écriture simplifiée :

```
lm(Y ~ . - X9 - X10, data = donnees)
```

Si nous souhaitons rajouter l'interaction entre `X1` et `X2` sans créer la variable produit `X1*X2` dans le data-frame, nous pouvons écrire :

```
lm(Y ~ X1 + X2 + X1:X2, data = donnees)
```

qui peut aussi s'écrire en version simplifiée :

```
lm(Y ~ X1*X2, data = donnees)
```

Si nous souhaitons étudier le modèle avec toutes les interactions d'ordre 2, nous écrivons :

```
lm(Y ~ .^2, data = donnees)
```

et de façon identique avec les interactions d'ordre supérieur. Nous allons présenter différents modèles dans la suite de ce chapitre.

6.2 Analyse de la covariance

6.2.1 Introduction : exemple des eucalyptus

L'analyse de la hauteur des arbres en fonction de la circonférence et de la provenance est un exemple classique d'analyse de la covariance. Afin de la mener à bien, il faut introduire la variable `bloc`.

La démarche la plus naturelle consiste à effectuer trois régressions différentes, une pour chaque champ, cela donne en termes de modélisation

$$\begin{array}{rcll} y_{i,A1} & = & \alpha_{A1} + \gamma_{A1} x_{i,A1} + \varepsilon_{i,A1}, & i = 1, \ldots, 527, \quad \text{bloc } A1 \\ y_{i,A2} & = & \alpha_{A2} + \gamma_{A2} x_{i,A2} + \varepsilon_{i,A2}, & i = 1, \ldots, 586, \quad \text{bloc } A2 \\ y_{i,A3} & = & \alpha_{A3} + \gamma_{A3} x_{i,A3} + \varepsilon_{i,A3}, & i = 1, \ldots, 316, \quad \text{bloc } A3 \end{array}$$

ou de manière simplifiée

$$y_{ij} = \alpha_j + \gamma_j x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3. \tag{6.1}$$

Pour chaque modèle, il suffit d'effectuer une régression simple.

Cependant, imaginons que nous savons que la circonférence intervient de la même façon dans chaque parcelle, c'est-à-dire que la pente est identique d'un champ à un autre. Les droites de régression sont donc parallèles. Cela donne graphiquement :

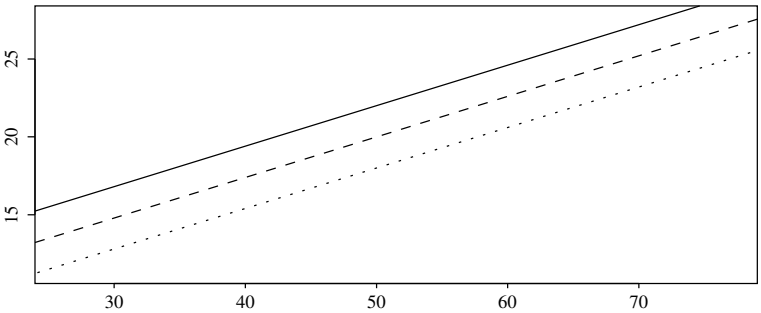


Fig. 6.2 – 3 droites de régression fictives parallèles.

et en termes de modélisation

$$\begin{array}{rcll} y_{i,A1} & = & \alpha_{A1} + \gamma\ x_{i,A1} + \varepsilon_{i,A1}, & i = 1, \ldots, 527, \quad \text{bloc A1} \\ y_{i,A2} & = & \alpha_{A2} + \gamma\ x_{i,A2} + \varepsilon_{i,A2}, & i = 1, \ldots, 586, \quad \text{bloc A2} \\ y_{i,A3} & = & \alpha_{A3} + \gamma\ x_{i,A3} + \varepsilon_{i,A3}, & i = 1, \ldots, 316, \quad \text{bloc A3.} \end{array}$$

Nous pouvons écrire de manière simplifiée

$$y_{ij} = \alpha_j + \gamma\ x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3. \tag{6.2}$$

Si nous savons que l'ordonnée à l'origine est la même pour chaque parcelle et que seule la pente change, nous obtenons graphiquement :

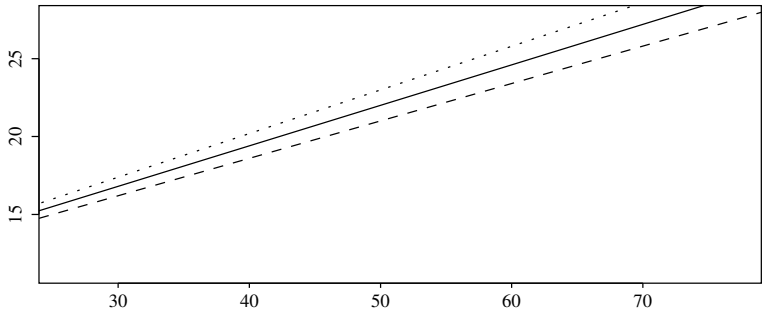


Fig. 6.3 – 3 droites de régression fictives ayant la même ordonnée à l'origine.

et en termes de modélisation

$$y_{ij} \quad = \quad \alpha + \gamma_j x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3. \tag{6.3}$$

Le coefficient γ dans le modèle (6.2) est le même dans tous les blocs. Si nous effectuons trois régressions distinctes, comment trouverons-nous la même estimation de γ ? De même, comment allons-nous procéder pour obtenir le même estimateur de α dans chaque population en effectuant trois régressions distinctes dans le modèle (6.3) ? Il semble raisonnable de n'effectuer qu'une seule régression mais avec des coefficients qui peuvent différer (ou non) selon les blocs.

6.2.2 Modélisation du problème

Nous traitons dans cette section le cas simple où nous disposons de deux variables explicatives : une variable quantitative notée X (dans l'exemple de l'eucalyptus X correspond à `circ`) et une variable qualitative notée A admettant I modalités dont le codage disjonctif est noté A_c (dans l'exemple de l'eucalyptus, A correspond à `bloc`, elle admet 3 modalités, A_c est ainsi une matrice de taille 1429×3). Nous notons X_c la matrice composée de n lignes et I colonnes où la j^{e} colonne de X_c correspond à la valeur de X lorsque les individus appartiennent à la modalité j, cela correspond au produit terme à terme de X avec chaque colonne de A_c.

$$circ = X = \begin{bmatrix} 36 \\ 42 \\ 38 \\ 46 \\ 36 \\ 45 \end{bmatrix} \quad \Longrightarrow \quad X_c = \begin{bmatrix} 36 & 0 & 0 \\ 42 & 0 & 0 \\ 0 & 38 & 0 \\ 0 & 46 & 0 \\ 0 & 0 & 36 \\ 0 & 0 & 45 \end{bmatrix}.$$

La matrice X_c correspond à l'interaction entre X et A. Pour chaque niveau j de la variable qualitative, nous observons n_j individus et les valeurs correspondantes de la variable X sont notées $x_{1j}, \ldots, x_{n_j j}$. De la même manière, nous notons les valeurs de la variable à expliquer $y_{1j}, \ldots, y_{n_j j}$. Le nombre total d'observations vaut $n = \sum_{i=1}^{I} n_i$.
Ecrivons matriciellement les trois modélisations proposées.

1. Soit nous considérons pour chaque niveau de la variable qualitative un modèle de régression (modèle 6.1), cela revient à analyser l'interaction entre les variables X et A, le modèle s'écrit alors

$$Y = A_c\alpha + X_c\gamma + \varepsilon. \tag{6.4}$$

Nous avons 7 paramètres à estimer (α et γ sont des vecteurs à 3 coordonnées) et σ est un scalaire positif correspondant à l'écart-type du bruit.

2. Soit nous considérons que la variable X intervient de la même façon quels que soient les niveaux de la variable A (la pente de la droite est toujours la même) et la variable A intervient seulement sur le niveau (l'ordonnée à l'origine de la droite de régression). Le modèle s'écrit alors

$$Y = A_c\alpha + X\gamma + \varepsilon. \tag{6.5}$$

Nous avons 5 paramètres à estimer (α est un vecteur à 3 coordonnées) et γ et σ sont des scalaires. Remarquons qu'ici l'interaction avec A ne se fait plus avec X, les pentes étant identiques. Cependant les ordonnées à l'origine étant différentes selon les niveaux de A, il subsiste une interaction entre A et la variable $\mathbb{1}$ de la régression (appelée en anglais et dans les logiciels `intercept`).

3. Soit nous considérons que la variable A intervient uniquement sur la pente et donc que l'ordonnée à l'origine ne change pas. Le modèle s'écrit

$$Y = \mathbb{1}\alpha + X_c\gamma + \varepsilon. \tag{6.6}$$

Nous avons 5 paramètres à estimer (γ est un vecteur à 3 coordonnées) et α et σ sont des scalaires.

Le choix du modèle (6.4) ou (6.5) ou (6.6) dépend du problème posé. Nous préconisons de commencer par le modèle le plus général (6.4) puis, si les pentes sont les mêmes, de passer au modèle simple (6.5) ou, si les ordonnées à l'origine sont les mêmes, de passer au modèle simple (6.6). Les modèles étant emboîtés, il est possible de tester un modèle contre un autre.
En pratique, avant d'effectuer une modélisation, il est préférable de représenter le nuage des points (X, Y) en couleur, où chaque couleur représente une modalité de la variable A. Cette représentation permet de se faire une idée des effets respectifs des différentes variables.

Remarque
Si nous additionnons toutes les colonnes de A_c nous obtenons le vecteur $\mathbb{1}$, la matrice $(\mathbb{1}, A_c)$ n'est pas de plein rang. De la même manière si nous additionnons toutes les colonnes de X_c nous obtenons la variable X, la matrice (X, X_c) n'est pas de plein rang. Dans ces cas, l'hypothèse $\mathcal{H}_1$ n'est pas vérifiée. Le projeté $\hat{Y}$ sur l'espace engendré par les colonnes de $(\mathbb{1}, A_c, X, X_c)$ existe, est unique mais son écriture en fonction des vecteurs (vecteurs colonnes) engendrant l'espace ne l'est pas. Nous aborderons dans la partie analyse de la variance de ce chapitre les différentes manières de procéder.

Les trois modèles que nous venons de voir peuvent s'écrire de manière générique

$$Y \quad = \quad X\beta + \varepsilon$$

où X est de taille respective $n \times 2I$ (6.4), et $n \times (I+1)$ dans les autres cas. Nous avons la propriété suivante (voir exercice 6.2) :

Proposition 6.1
L'estimateur des MC de β est obtenu dans le modèle (6.4) en effectuant une régression simple pour chaque niveau i de la variable qualitative A. L'estimateur des MC de σ^2 est

$$\hat{\sigma}^2 = \frac{1}{n-2I} \sum_{j=1}^{I} \sum_{i=1}^{n_i} (y_{ij} - \hat{y}_{ij})^2 .$$

Remarquons que, même si l'estimateur des MC de β peut être obtenu en effectuant une régression simple pour chaque niveau i de la variable A, l'analyse de la covariance suppose l'égalité des variances des erreurs pour chacun des niveaux i de la variable A. Il n'en va pas de même pour les I régressions simples où les modèles ne sont pas contraints à avoir la même variance et où l'on aura donc I variances d'erreurs différentes.

6.2.3 Hypothèse gaussienne

Sous l'hypothèse de normalité des erreurs ε, nous pouvons tester toutes les hypothèses linéaires possibles. Les modèles (6.5) et (6.6) sont emboîtés dans le modèle général (6.4). Un des principaux objectifs de l'analyse de la covariance est de *savoir si les variables explicatives influent sur la variable à expliquer*. Les deux premiers tests que nous effectuons sont

1. le test d'égalité des pentes

$$\mathrm{H}_0 \quad : \quad \gamma_1 = \cdots = \gamma_I = \gamma \qquad \mathrm{H}_1 \quad : \quad \exists (i,j) : \gamma_i \neq \gamma_j$$

Cela revient à tester le modèle (6.5) contre (6.4).

2. Le test d'égalité des ordonnées à l'origine

$$\mathrm{H}_0 \quad : \quad \alpha_1 = \cdots = \alpha_I = \alpha \qquad \mathrm{H}_1 \quad : \quad \exists (i,j) : \alpha_i \neq \alpha_j$$

Cela revient à tester le modèle (6.6) contre (6.4).

La statistique de test vaut donc (théorème 5.2, p. 100)

$$F = \frac{\|\hat{Y} - \hat{Y}_0\|^2/(I-1)}{\|Y - \hat{Y}\|^2/(n-2I)}.$$

L'hypothèse H_0 sera rejetée en faveur de H_1 si l'observation de la statistique F est supérieure à $f_{I-1,n-I}(1-\alpha)$ et nous conclurons à l'effet du facteur explicatif.

Pour résumer, nous partons donc du modèle complet

$$Y = A_c\alpha + X_c\gamma + \varepsilon,$$

et acceptons :
— soit

$$Y = A_c\alpha + X\gamma + \varepsilon,$$

nous pouvons ensuite tester soit la nullité de la pente (la variable quantitative X n'apporte pas d'information quant à l'explication de Y), soit l'égalité des différentes α_i (la variable qualitative A n'apporte pas d'information quant à l'explication de Y) ;
— soit

$$Y = \mathbb{1}\alpha + X_c\gamma + \varepsilon,$$

nous pouvons ensuite tester l'égalité des pentes (la variable qualitative A n'apporte pas d'information quant à l'explication de Y).

6.2.4 Exemple : la concentration en ozone

Nous souhaitons expliquer la concentration en ozone `O3` en fonction de la température `T12` et de la direction du vent `vent`, variable qualitative prenant 4 modalités : `NORD`, `SUD`, `EST` et `OUEST`. Le modèle que nous souhaitons estimer est le suivant :

$$ozone = \alpha + \beta T_{12} + \gamma vent + \varepsilon.$$

La variable `vent` étant qualitative, elle doit être codée et le modèle s'écrit

$$ozone = \alpha + \beta T_{12} + \gamma_E 1_{vent=NORD} + \cdots + \gamma_O 1_{vent=OUEST} + \varepsilon.$$

Ce modèle n'est pas identifiable et nous allons voir les différentes manières de poser des contraintes identifiantes.
Nous commençons cette étude par l'analyse graphique (fig. 6.4).

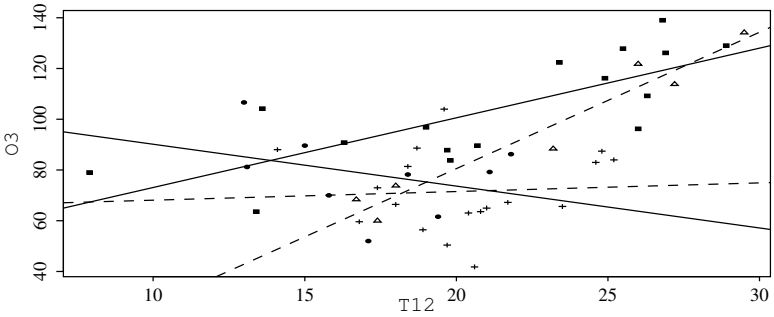


Fig. 6.4 – Nuage de points et régression simple pour chaque niveau de vent. Le niveau de vent est représenté par un symbole (rond, triangle, +, carré) différent.

Les pentes des différentes régressions sont différentes, il semble que la modélisation de la concentration de l'ozone en fonction de la température dépende de la variable vent. Pour obtenir le graphique (6.4), nous utilisons les commandes suivantes :

```
> ozone <- read.table("ozone.txt", header = T, sep = ";")
> plot(ozone[,"T12"], ozone[,"O3"],pch=as.numeric(ozone[,"vent"]),
+       col = as.numeric(ozone[,"vent"]))
> a1 <- lm(O3 ~ T12, data = ozone[ozone[,"vent"]=="EST",])
> a2 <- lm(O3 ~ T12, data = ozone[ozone[,"vent"]=="NORD",])
> a3 <- lm(O3 ~ T12, data = ozone[ozone[,"vent"]=="OUEST",])
> a4 <- lm(O3 ~ T12, data = ozone[ozone[,"vent"]=="SUD",])
> abline(a1, col=1)
> abline(a2, col=2)
> abline(a3, col=3)
> abline(a4, col=4)
```

Le modèle avec interaction (6.4) s'écrit :

```
> mod1b <- lm(formula = O3 ~ -1 + vent + T12:vent, data = ozone)
```

Nous enlevons la constante en écrivant -1. Ensuite il faut conserver une ordonnée à l'origine différente pour chacune des modalités du vent, ce qui est représenté par le facteur `vent` (ou une interaction de la variable `1` avec `vent`). Ensuite nous ajoutons un coefficient directeur différent pour chacune des modalités du vent, ce qui est représenté par la variable `T12` en interaction avec `vent`. Cela donne :

```
> summary(mod1b)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
ventEST        45.6090    13.9343   3.273 0.002133 **
ventNORD      106.6345    28.0341   3.804 0.000456 ***
ventOUEST      64.6840    24.6208   2.627 0.011967 *
ventSUD       -27.0602    26.5389  -1.020 0.313737
ventEST:T12     2.7480     0.6342   4.333 8.96e-05 ***
ventNORD:T12   -1.6491     1.6058  -1.027 0.310327
ventOUEST:T12   0.3407     1.2047   0.283 0.778709
ventSUD:T12     5.3786     1.1497   4.678 3.00e-05 ***
```

Si, dans l'écriture du modèle, la constante est conservée, le logiciel va prendre comme cellule de référence la première cellule (définie par ordre lexicographique). Cela donne :

```
> mod1 <- lm(formula = O3 ~ vent + T12:vent, data = ozone)
> summary(mod1)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    45.6090    13.9343   3.273  0.00213 **
ventNORD       61.0255    31.3061   1.949  0.05796 .
ventOUEST      19.0751    28.2905   0.674  0.50384
ventSUD       -72.6691    29.9746  -2.424  0.01972 *
ventEST:T12     2.7480     0.6342   4.333 8.96e-05 ***
ventNORD:T12   -1.6491     1.6058  -1.027  0.31033
ventOUEST:T12   0.3407     1.2047   0.283  0.77871
ventSUD:T12     5.3786     1.1497   4.678 3.00e-05 ***
```

Les coefficients des ordonnées à l'origine sont des effets différentiels par rapport à la cellule de référence (ici ventEST). Par exemple 61.0255+45.6090=106.6345 qui est la valeur de ventNord dans l'écriture précédente.

Le modèle avec une seule pente (6.5) peut s'écrire :

```
> mod2 <- lm(formula = O3 ~ vent + T12, data = ozone)
> mod2b <- lm(formula = O3 ~ -1 + vent + T12, data = ozone)
```

Le modèle avec une seule ordonnée à l'origine (6.6) peut s'écrire :

```
> mod3 <- lm(formula = O3 ~ vent:T12, data = ozone)
```

Nous procédons de la manière suivante pour choisir la meilleure modélisation.

1. **Egalité des pentes** : nous effectuons un test entre le modèle (6.5) et (6.4) grâce à la commande :

```
> anova(mod2,mod1)
Analysis of Variance Table
Model 1: O3 ~ T12 + vent
Model 2: O3 ~ vent + T12:vent
  Res.Df     RSS Df Sum of Sq      F   Pr(>F)
1     45 12612.0
2     42  9087.4  3    3524.5 5.4298 0.003011 **
```

Nous concluons donc à l'effet du vent sur les pentes comme nous le suggérait la figure 6.4. Nous aurions obtenu les mêmes résultats avec mod2b contre mod1, ou mod2 contre mod1b ou encore mod2b contre mod1b.

2. **Egalité des ordonnées à l'origine** : nous effectuons un test entre le modèle (6.6) et (6.4) grâce à la commande :

```
> anova(mod3,mod1)
Analysis of Variance Table
Model 1: O3 ~ vent:T12
Model 2: O3 ~ vent + T12:vent
  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1     45 11864.1
2     42  9087.4  3    2776.6 4.2776 0.01008 *
```

Nous concluons donc à l'effet du vent sur les ordonnées à l'origine comme nous le suggérait la figure 6.4.

Enfin, le graphique de résidus (fig. 6.5) obtenu avec :

```
> plot(rstudent(mod2) ~ fitted(mod2),xlab="ychap",ylab="residus")
```

ne fait apparaître ni structure ni point aberrant.

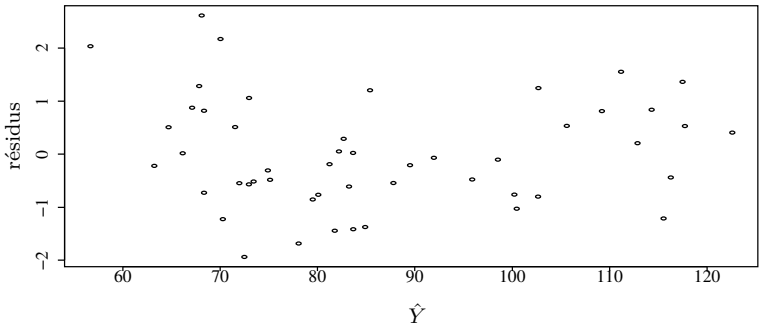


Fig. 6.5 – Résidus studentisés du modèle 1.

En revanche, si on analyse la structure des résidus par modalité de `Vent`

```
> xyplot(rstudent(mod2)~fitted(mod2)|vent,
                              data = ozone, ylab="residus")
```

on constate une structuration des résidus pour la modalité `SUD`. Cependant cette structuration n'est constatée qu'avec 7 individus, ce qui semble trop peu pour que cette conclusion soit fiable.

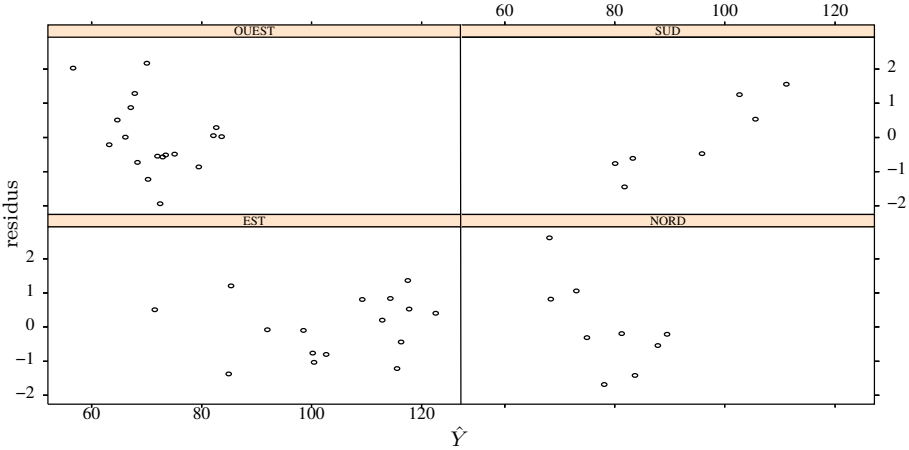


Fig. 6.6 – Résidus studentisés du modèle 1 (ou 1b) par niveau de vent.

Remarque
Pour l'exemple de l'ozone, nous conservons donc le modèle complet. Il faut faire attention à l'écriture du modèle en langage « logiciel ». L'écriture logique du point de vue du logiciel consiste à écrire :

```
> mod <- lm(formula = O3 ~ vent + T12 + T12:vent, data = ozone)
```

En effet, nous utilisons bien les 3 variables `vent`, `T12` et leur interaction. En écrivant de cette manière, la matrice X du modèle est composée de $\mathbb{1}$, A_c, `T12` et de `T12`$_c$. Cette matrice n'est pas de plein rang. Le logiciel, pour pouvoir inverser cette matrice, doit imposer des contraintes (que nous verrons plus en détail dans la suite de ce chapitre). Le logiciel R va prendre comme cellule de référence la première cellule [3] (définie par ordre lexicographique) et calculer des effets différentiels par rapport à cette cellule. Sur l'exemple de l'ozone la cellule de référence va être `EST` et nous obtenons :

3. Dans certaines procédures, SAS utilise la dernière cellule comme cellule de référence.

```
> mod0 <- lm(formula = O3 ~ vent +T12 + T12:vent, data = ozone)
> summary(mod0)
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     45.6090    13.9343   3.273  0.00213 **
ventNORD        61.0255    31.3061   1.949  0.05796 .
ventOUEST       19.0751    28.2905   0.674  0.50384
ventSUD        -72.6691    29.9746  -2.424  0.01972 *
T12              2.7480     0.6342   4.333 8.96e-05 ***
ventNORD:T12    -4.3971     1.7265  -2.547  0.01462 *
ventOUEST:T12   -2.4073     1.3614  -1.768  0.08429 .
ventSUD:T12      2.6306     1.3130   2.004  0.05160 .
```

`Intercept` et `T12` sont bien les valeurs de l'ordonnée à l'origine et de la pente pour le vent d'EST.

6.2.5 Exemple : la hauteur des eucalyptus

Nous commençons par le modèle complet obtenu grâce aux commandes :

```
> eucalypt[,"bloc"] <- as.factor(eucalypt[,"bloc"])
> m.complet <- lm(ht ~ bloc - 1 + bloc:circ, data = eucalypt)
```

qui correspond au modèle

$$y_{ij} \quad = \quad \alpha_j + \gamma_j x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3.$$

Nous estimons ensuite les paramètres dans le modèle admettant une pente commune quelle que soit l'origine des eucalyptus

$$y_{ij} \quad = \quad \alpha_j + \gamma \; x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3$$

grâce à la commande :

```
> m.pente <- lm(ht ~ bloc - 1 + circ, data = eucalypt)
```

Nous estimons également les paramètres dans le modèle où nous supposons que l'origine de l'arbre influence la pente uniquement

$$y_{ij} \quad = \quad \alpha + \gamma_j x_{ij} + \varepsilon_{ij} \quad i = 1, \ldots, n_j \quad \text{champ} \quad j \quad j = A1, A2, A3$$

avec la commande

```
> m.ordonne <- lm(ht ~ bloc:circ, data = eucalypt)
```

Les deux derniers modèles sont emboîtés dans le premier. Nous pouvons tester :

1. **L'égalité des pentes**

```
> anova(m.pente, m.complet)
Analysis of Variance Table
Model 1: ht ~ bloc - 1 + circ
Model 2: ht ~ bloc - 1 + bloc:circ
  Res.Df     RSS   Df Sum of Sq      F Pr(>F)
1   1425 2005.90
2   1423 2005.05    2      0.85 0.3007 0.7403
```

Nous conservons le modèle avec une seule pente.

2. **L'égalité des ordonnées**

```
> anova(m.ordonne, m.complet)
Analysis of Variance Table
Model 1: ht ~ bloc:circ
Model 2: ht ~ bloc - 1 + bloc:circ
  Res.Df     RSS   Df Sum of Sq      F Pr(>F)
1   1425 2009.21
2   1423 2005.05    2      4.16 1.4779 0.2285
```

Nous conservons le modèle avec une seule ordonnée à l'origine.

Nous avons donc le choix entre les 2 modèles

$$\begin{aligned} y_{ij} &= \alpha + \gamma_j x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3 \\ y_{ij} &= \alpha_j + \gamma x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3. \end{aligned}$$

Ces modèles ne sont pas emboîtés. Cependant nous estimons le même nombre de paramètres (4) et nous pouvons donc comparer ces modèles *via* leur R^2. Nous choisissons le modèle avec une pente. Pour terminer cette étude, nous comparons le modèle retenu avec le modèle de régression simple, c'est-à-dire le modèle où l'origine n'intervient pas :

$$y_{ij} = \alpha + \gamma x_{ij} + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = A1, A2, A3.$$

```
> m.simple <- lm(ht ~ circ, data = eucalypt)
> anova(m.simple, m.pente)
Analysis of Variance Table

Model 1: ht ~ circ
Model 2: ht ~ bloc - 1 + circ
  Res.Df     RSS   Df Sum of Sq      F   Pr(>F)
1   1427 2052.08
2   1425 2005.90    2     46.19 16.406 9.03e-08 ***
```

Nous conservons le modèle avec des ordonnées différentes à l'origine selon le bloc mais une même pente. Pour terminer cette étude, analysons les résidus studentisés.

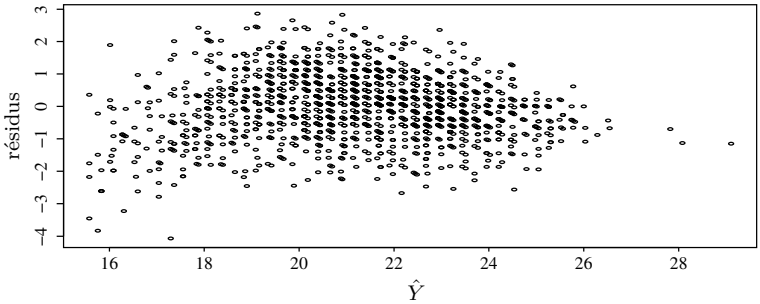


Fig. 6.7 – Résidus studentisés du modèle avec des pentes identiques.

6.3 Analyse de la variance à 1 facteur

6.3.1 Introduction

Nous modélisons la concentration d'ozone en fonction du vent (quatre secteurs donc quatre modalités). Dans le tableau suivant figurent les valeurs des 10 premiers individus du tableau de données.

individu	1	2	3	4	5	6	7	8	9	10
O_3	64	90	79	81	88	68	139	78	114	42
Vent	E	N	E	N	O	S	E	N	S	O

Tableau 6.2 – Tableau des données brutes.

La première analyse à effectuer est une représentation graphique des données. Les boîtes à moustaches (boxplots) de la variable Y par cellule semblent les plus adaptées à l'analyse.

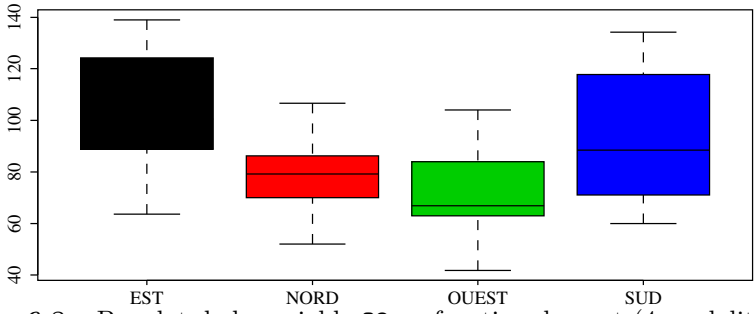


Fig. 6.8 – Boxplot de la variable `O3` en fonction du vent (4 modalités).

Au vu de ce graphique, il semblerait que le vent ait une influence sur la valeur de la concentration d'ozone. La concentration est plus élevée en moyenne lorsque le vent

vient de l'EST et au contraire moins élevée lorsque le vent vient de la mer (NORD et OUEST). Afin de préciser cette hypothèse, nous allons construire une analyse de la variance à un facteur explicatif : le vent.

6.3.2 Modélisation du problème

Dans ce cas simple, nous avons une variable explicative et une variable à expliquer et nous voulons expliquer la concentration d'ozone par le vent. Ce cas est appelé analyse de variance[4] à un facteur, qui est la variable qualitative explicative. Nous remplaçons la variable A par son codage disjonctif complet, c'est-à-dire que nous remplaçons le vecteur A par $I = 4$ vecteurs $\mathbb{1}_{\texttt{NORD}}, \mathbb{1}_{\texttt{SUD}}, \mathbb{1}_{\texttt{EST}}, \mathbb{1}_{\texttt{OUEST}}$ indiquant l'appartenance aux modalités NORD, SUD, EST ou OUEST. Ces quatre vecteurs sont regroupés dans la matrice $A_c = (\mathbb{1}_{\texttt{NORD}}, \mathbb{1}_{\texttt{SUD}}, \mathbb{1}_{\texttt{EST}}, \mathbb{1}_{\texttt{OUEST}})$. Le modèle de régression s'écrit alors sous forme matricielle

$$Y = \mu\mathbb{1} + A_c\alpha + \varepsilon. \tag{6.7}$$

La variable qualitative A engendre une partition des observations en I groupes (ici 4) souvent appelés cellules. La i^{e} cellule est constituée des n_i observations de la variable à expliquer Y admettant le caractère i de la variable explicative. Nous avons au total n observations avec $n = \sum_{i=1}^{I} n_i$. Les données sont ainsi regroupées en cellules selon le tableau suivant :

Vent	NORD	SUD	EST	OUEST
O_3	90	68	64	88
	81	114	79	42
	78		139	

Tableau 6.3 – Tableau des données brutes regroupées par cellule.

Classiquement, en analyse de la variance, on utilise des tableaux de la forme (6.3). Dans ce tableau, la notation des n individus ne se fait pas classiquement de 1 à n. En effet, doit-on lire l'ordre des individus dans le sens des lignes du tableau ou dans le sens des colonnes ? Par convention, la valeur y_{ij} correspond au j^{e} individu de la cellule i. Les individus ne seront donc plus numérotés de 1 à n mais suivant le schéma $(1,1)$, $(1,2)$, $\cdots$, $(1,n_1)$, $(2,1)$, $(2,2)$, $\cdots$, $(I,1)$, $\cdots$, (I,n_I) pour bien insister sur l'appartenance de l'individu à la modalité i qui varie de 1 à I.
Le modèle

$$y_i = \mu + \alpha_1 A_{1i} + \alpha_2 A_{2i} + \alpha_3 A_{3i} + \alpha_4 A_{4i} + \varepsilon_i, \quad i = 1, \ldots, n$$

s'écrit alors avec ces notations

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = 1, \ldots, 4. \tag{6.8}$$

4. Nous utilisons aussi l'acronyme ANOVA *(analysis of variance)* très répandu en statistique.

Revenons à l'écriture matricielle

$$Y = \mu \mathbb{1} + A_c \alpha + \varepsilon$$

Si nous additionnons toutes les colonnes de A_c nous obtenons le vecteur $\mathbb{1}$, la matrice $X = (\mathbb{1}, A_c)$ n'est pas de plein rang et l'hypothèse $\mathcal{H}_1$ n'est pas vérifiée. Remarquons que cela entraîne que $(\mathbb{1}, A_c)'(\mathbb{1}, A_c)$ n'est pas de plein rang et nous ne pouvons pas calculer son inverse directement. Nous ne pouvons donc pas appliquer directement au modèle (6.7) les résultats des chapitres précédents. Nous sommes ici confrontés à un problème d'identifiabilité.

Peut-on estimer μ et α ou plus exactement peut-on déterminer μ et α de manière unique ? En termes statistiques le modèle est-il identifiable ? Considérons le modèle (6.8) et posons $\widetilde{\mu} = \mu + 1024$ et $\widetilde{\alpha}_i = \alpha_i - 1024$ pour $i = 1, \ldots, I$, nous avons alors

$$y_{ij} = \widetilde{\mu} + \widetilde{\alpha}_i + \varepsilon_{ij} = \mu + \alpha_i + \varepsilon_{ij}.$$

Deux valeurs différentes des paramètres donnent les mêmes valeurs pour Y. Il est même facile de voir qu'une infinité de valeurs différentes de (μ, α_i) donnent le même Y. Ainsi, la loi de Y peut s'écrire avec une infinité de valeurs différentes des paramètres (μ, α_i), on ne peut donc pas identifier cette loi à un unique vecteur de paramètres. C'est en ce sens qu'on dit que le modèle est non identifiable. L'identifiabilité est nécessaire en statistique paramétrique. En effet, avant de chercher de bons estimateurs pour des paramètres, il est nécessaire que les paramètres que l'on cherche à estimer soient définis de façon unique.

Identifiabilité et contraintes

Afin de pallier les problèmes d'identifiabilité, la méthode la plus classique consiste à se donner des contraintes sur les paramètres. Pour d'autres approches le lecteur pourra se reporter au paragraphe 6.6. On considère le modèle 6.7. Pour résoudre les problèmes d'identifiabilité évoqués ci-dessus, nous proposons ici d'utiliser des contraintes linéaires de la forme $\sum_{j=1}^{I} a_j \alpha_j = 0$ où on rappelle que I désigne le nombre de valeurs possibles de la variable explicative qualitative et où les $a_j, j = 1, \ldots, I$ sont des réels à spécifier. On peut déjà remarquer qu'avec la contrainte linéaire, nous aurons besoin d'estimer uniquement $I - 1$ paramètres parmi $(\alpha_1, \ldots, \alpha_I)$. Le dernier paramètre se déduira de la contrainte.
Ces contraintes linéaires sont appelées contraintes identifiantes et voici les plus classiques en notation ANOVA :

— choisir un des $\alpha_i = 0$, la cellule i sert de cellule de référence (c'est ce que R fait généralement par défaut) ;
— choisir $\sum n_i \alpha_i = 0$, la contrainte d'orthogonalité. Lorsque le plan est équilibré (les n_i sont tous égaux), cette contrainte devient $\sum \alpha_i = 0$;
— choisir $\sum \alpha_i = 0$, contrainte parfois utilisée par certains logiciels. Cette contrainte représente l'écart au coefficient constant μ. Remarquons toutefois qu'à l'image

de la régression simple, le coefficient constant μ n'est en général pas estimé par la moyenne empirique générale $\bar{y}$ sauf si le plan est équilibré.

Une autre contrainte, qui n'appartient pas à la famille des contraintes linéaires présentée ci-dessus, consiste à choisir $\mu = 0$: on supprime la constante μ du modèle et il nous faut alors estimer un paramètre par niveau du facteur (voir modèle 6.9 ci-dessous).

6.3.3 Interprétation des contraintes

Il est intéressant de visualiser ces différentes modélisations sur un graphique (fig. 6.9). Pour ce faire, nous considérons un facteur admettant deux modalités. Nous avons donc le modèle d'ANOVA suivant

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij}.$$

Ce modèle étant non-identifiable, nous pouvons réécrire ce modèle en remplaçant $\mu + \alpha_1$ par m_1 et $\mu + \alpha_2$ par m_2 :

$$y_{ij} = m_i + \varepsilon_{ij}. \tag{6.9}$$

Les contraintes identifiantes sont représentées graphiquement dans la figure 6.9.

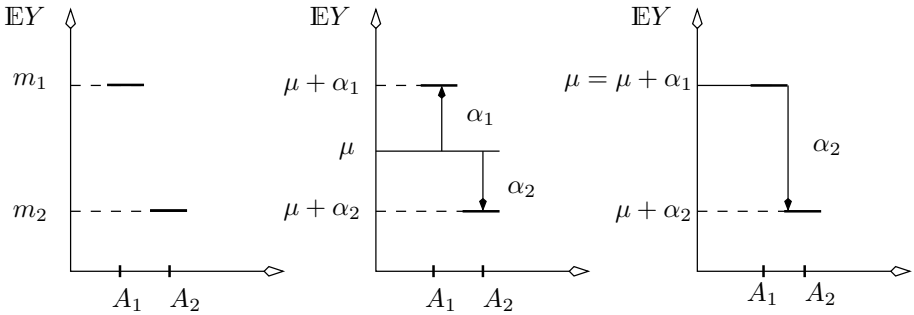


Fig. 6.9 – Modélisations selon les contraintes sur les paramètres.

La premier graphique à gauche représente les espérances de Y pour chaque niveau du facteur (ou autrement dit dans chaque cellule), espérances notées m_1 et m_2, ce qui correspond à $\mu = 0$. Le deuxième graphique représente la contrainte $\sum_i \alpha_i = 0$. Rappelons que si le plan est équilibré, cette contrainte revient à $\sum_i n_i \alpha_i = 0$. Ici μ représente la « moyenne générale » et les α sont les effets différentiels. Le troisième graphique représente la contrainte $\alpha_i = 0$, une cellule (i.e. un niveau du facteur) est prise comme cellule de référence.

6.3.4 Estimation des paramètres

La proposition suivante présente les estimateurs des moindres carrés des paramètres du modèle d'analyse de variance à un facteur en fonction de différents types de contraintes.

Proposition 6.2
Soit le modèle d'analyse de la variance à un facteur

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij}. \tag{6.10}$$

1. Sous la contrainte $\mu = 0$, qui correspond à $y_{ij} = \alpha_i + \varepsilon_{ij}$, les estimateurs des moindres carrés des paramètres inconnus sont :

$$\hat{\alpha}_i = \bar{y}_i, \quad avec \quad \bar{y}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} y_{ij}.$$

Les $\hat{\alpha}_i$ correspondent à la moyenne de la cellule.

2. Sous la contrainte $\alpha_1 = 0$, qui correspond à $y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$, les estimateurs des moindres carrés des paramètres inconnus sont :

$$\hat{\mu} = \bar{y}_1 \quad et \quad \hat{\alpha}_i = \bar{y}_i - \bar{y}_1.$$

La première cellule sert de référence. Le coefficient $\hat{\mu}$ est donc égal à la moyenne empirique de la cellule de référence, les $\hat{\alpha}_i$ correspondent à l'effet différentiel entre la moyenne de la cellule i et la moyenne de la cellule de référence.

3. Sous la contrainte $\sum n_i\alpha_i = 0$, qui correspond à $y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$, les estimateurs des moindres carrés des paramètres inconnus sont :

$$\hat{\mu} = \bar{y}, \quad \hat{\alpha}_i = \bar{y}_i - \bar{y} \quad et \quad \bar{y} = \sum_{i=1}^{I}\sum_{j=1}^{n_i} y_{ij}.$$

L'estimateur de la constante, noté $\hat{\mu}$, est donc la moyenne générale. Les $\hat{\alpha}_i$ correspondent à l'effet différentiel entre la moyenne de la cellule i et la moyenne générale.

4. Sous la contrainte $\sum \alpha_i = 0$, qui correspond à $y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$, les estimateurs des moindres carrés des paramètres inconnus sont :

$$\hat{\mu} = \bar{\bar{y}} = \frac{1}{I}\sum_{i=1}^{I} \bar{y}_i \quad et \quad \hat{\alpha}_i = \bar{y}_i - \bar{\bar{y}}.$$

Les $\hat{\alpha}_i$ correspondent à l'effet différentiel entre la moyenne empirique de la cellule i et la moyenne des moyennes empiriques. Lorsque le plan est déséquilibré, les α_i sont toujours les écarts à μ, cependant ce dernier n'est pas estimé par la moyenne générale empirique, mais par la moyenne des moyennes empiriques.

Dans tous les cas, σ^2 est estimé par :

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{I}\sum_{j=1}^{n_i}(y_{ij} - \bar{y}_i)^2}{n - I}.$$

La preuve est à faire en exercice (voir exercice 6.3).

6.3.5 Hypothèse gaussienne et test d'influence du facteur

Afin d'établir des intervalles de confiance ou des procédures de test pour les différents paramètres, nous introduisons à nouveau l'hypothèse de normalité des erreurs ε, notée $\mathcal{H}_3 : \varepsilon \sim \mathcal{N}(0, \sigma^2)$.
Un des principaux objectifs de l'analyse de la variance est de *savoir si le facteur possède une influence sur la variable à expliquer*. Dire que le facteur n'a pas d'influence signifie que la loi de Y est la même pour toutes les valeurs du facteur. Dans le modèle d'analyse de variance, cela revient à dire que toutes les valeurs α_i sont identiques. Les hypothèses du test seront alors :

$$\mathrm{H}_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_I \quad \text{contre} \quad \mathrm{H}_1 : \exists (i,j) \text{ tel que } \alpha_i \neq \alpha_j.$$

Remarquons que si l'on choisit une contrainte linéaire sur les α_i alors l'hypothèse H_0 est équivalente à $\mathrm{H}_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_I = 0$. Le modèle sous H_0 peut s'écrire encore sous la forme suivante $y_{ij} = \mu + \varepsilon_{ij}$ avec $\varepsilon_{ij} \sim \mathcal{N}(0, \sigma_0^2)$. Nous sommes en présence d'un test entre deux modèles dont l'un est un cas particulier de l'autre (voir section 5.5.2, p. 98). La statistique de test vaut donc (Théorème 5.2 p. 100)

$$F = \frac{\|\hat{Y} - \hat{Y}_0\|^2/(I-1)}{\|Y - \hat{Y}\|^2/(n-I)}.$$

où $\hat{Y}_0$ la projection orthogonale de Y sur la constante $\mathbb{1}$. Rappelons, même si la valeur de σ_0^2 n'est pas utile pour le test, que sous H_0 les estimateurs sont

$$\hat{\mu} = \bar{y} \quad \text{et} \quad \hat{\sigma}_0^2 = \frac{1}{n-1} \sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y})^2.$$

Les termes de la statistique de test s'écrivent alors

$$\|\hat{Y} - \hat{Y}_0\|^2 = \sum_{i=1}^{I} n_i (\bar{y}_i - \bar{y})^2, \tag{6.11}$$

$$\|Y - \hat{Y}\|^2 = \sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2. \tag{6.12}$$

Pour tester l'influence de la variable explicative, nous avons le théorème suivant :

Théorème 6.1
Soit le modèle d'analyse de la variance à 1 facteur (6.10) *muni d'une contrainte linéaire sur les* α_i *(*$\alpha_1 = 0$ *ou* $\sum n_i \alpha_i = 0$ *ou* $\sum \alpha_i = 0$*). Notons l'hypothèse nulle (modèle restreint)* $\mathrm{H}_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_I = 0$ *qui correspond au modèle* $y_{ij} = \mu + \varepsilon_{ij}$ *et l'hypothèse alternative (modèle complet)* $\mathrm{H}_1 : \exists (i,j)$ *tel que* $\alpha_i \neq \alpha_j$ *qui correspond au modèle complet* $y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$.
Pour tester ces deux hypothèses nous utilisons la statistique de test ci-dessous qui possède comme loi sous H_0 *:*

$$F = \frac{\sum_{i=1}^{I} n_i (\bar{y}_i - \bar{y})^2}{\sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2} \times \frac{n-I}{I-1} \sim \mathcal{F}_{I-1, n-I}.$$

L'hypothèse H_0 *sera rejetée en faveur de* H_1 *au niveau* α *si l'observation de la statistique* F *est supérieure à* $f_{I-1,n-I}(1-\alpha)$ *et nous conclurons alors à l'effet du facteur explicatif.*

La preuve de ce théorème se fait facilement. Il suffit d'appliquer le théorème 5.2 p. 100 avec l'écriture des normes données en (6.11) et (6.12). Ces résultats sont en général résumés dans un tableau dit tableau d'analyse de la variance.

variation	ddl	SC	CM	valeur du F	$\Pr(> F)$
facteur	$I-1$	$\mathrm{SCA} = \|\hat{Y} - \hat{Y}_0\|^2$	$\mathrm{CMA} = \dfrac{\mathrm{SCA}}{(I-1)}$	$\dfrac{\mathrm{CMA}}{\mathrm{CMR}}$	
résiduelle	$n-I$	$\mathrm{SCR} = \|Y - \hat{Y}\|^2$	$\mathrm{CMR} = \dfrac{\mathrm{SCR}}{(n-I)}$		

Tableau 6.4 – Tableau d'analyse de la variance.

La première colonne indique la source de la variation, la seconde le degré de liberté associé à chaque effet. La somme des carrées (SCR) est rappelée dans le tableau ainsi que le carré moyen (CM) qui par définition est la SCR divisée par le ddl.

Conclusion

— En général, lors d'une analyse de la variance, nous nous intéressons à vérifier si le facteur a un effet sur Y. Nous répondons à cette question en testant l'égalité des paramètres associés aux modalités du facteur (ce qui suppose l'hypothèse de normalité). Le tableau d'analyse de variance permet d'effectuer le test et de conclure.

— Il faut représenter les résidus estimés afin de vérifier les hypothèses, notamment la normalité des erreurs. Une attention particulière sera portée à *l'égalité des variances dans les cellules*. Les tests F utilisés sont relativement robustes à la non normalité dans le cas où la distribution est unimodale et peu dissymétrique.

— Une investigation plus fine peut être ensuite entreprise en testant des hypothèses particulières comme la nullité de certains niveaux du facteur. Bien évidemment, après avoir choisi une contrainte identifiante, nous pouvons nous intéresser aux coefficients eux-mêmes en conservant à l'esprit que le choix de la contrainte a une influence sur la valeur des estimateurs.

6.3.6 Exemple : la concentration en ozone

Nous reprenons l'exemple introduction où nous souhaitons expliquer la concentration en ozone par la variable `vent` à 4 modalités à l'aide du modèle d'ANOVA

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij}, \quad i = 1, \ldots, n_j, \quad j = 1, \ldots, 4.$$

Nous mettons en œuvre sur R les différentes contraintes d'identifiabilité que nous venons de présenter.

1. $\mu = 0$. Pour obtenir cette contrainte, il suffit de spécifier au logiciel un modèle sans `intercept`

```
> mod1 <- lm(O3~vent-1,data=ozone)
```

Si nous souhaitons quantifier les effets des modalités, nous examinons les coefficients.

```
> summary(mod1)
Coefficients:
          Estimate Std. Error t value Pr(>|t|)
ventEST    103.850      4.963   20.92  < 2e-16 ***
ventNORD    78.289      6.618   11.83 1.49e-15 ***
ventOUEST   71.578      4.680   15.30  < 2e-16 ***
ventSUD     94.343      7.504   12.57  < 2e-16 ***
```

Nous obtenons bien comme estimateur de chaque paramètre la moyenne empirique de la teneur en O3 dans chaque groupe. Il faut faire attention au listing lorsque la constante n'est pas dans le modèle, ainsi pour le calcul du R^2 le logiciel utilise la formule sans constante. En général, lors d'une analyse de la variance, nous ne sommes pas intéressés par le test admettant comme hypothèse $H_0 : \alpha_i = 0$ et donc les dernières colonnes du listing ne sont pas d'un grand intérêt. Nous sommes intéressés par la question suivante : y a-t-il une influence du vent sur la concentration en O3 ? Pour répondre à cette question, R propose la fonction **anova**, que nous avons déjà utilisée dans la section précédente, et qui permet de tester des modèles emboîtés. Si cette fonction est utilisée avec un seul modèle, il faut que la constante soit dans le modèle. Quand la constante ne fait pas partie du modèle, le test effectué n'a pas trop de sens puisque l'on effectue un test entre le modèle à un facteur et le modèle $y_{ij} = 0 + \varepsilon_{ij}$. Ainsi dans l'exemple précédent nous avons :

```
> anova(mod1)
Analysis of Variance Table
Response: O3
          Df Sum Sq Mean Sq F value    Pr(>F)
vent       4 382244   95561  242.44 < 2.2e-16 ***
Residuals 46  18131     394
```

Pour savoir s'il y a un effet vent dans le cas de l'analyse à un facteur, il faut utiliser les autres contraintes comme nous allons le voir.

2. $\alpha_1 = 0$. Le logiciel R utilise par défaut la contrainte $\alpha_1 = 0$ appelée contraste « *treatment* ». Cela revient dans notre cas à prendre la cellule EST comme cellule de référence (la première par ordre alphabétique). La commande pour effectuer l'analyse est :

```
> mod2 <- lm(O3 ~ vent, data = ozone)
```

Pour répondre à la question sur l'influence du vent sur la concentration, nous analysons le tableau d'analyse de la variance donné par :

```
> anova(mod2)
Analysis of Variance Table
Response: O3
             Df  Sum Sq Mean Sq F value    Pr(>F)
vent          3  9859.8  3286.6  8.3383 0.0001556 ***
Residuals    46 18131.4   394.2
```

L'hypothèse H_0 est rejetée. En conclusion, il existe un effet vent. Si nous nous intéressons aux coefficients, ceux-ci sont différents du modèle `mod1` puisque nous avons changé la formulation du modèle. Examinons-les grâce à la commande suivante

```
> summary(mod2)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  103.850      4.963  20.923  < 2e-16 ***
ventNORD     -25.561      8.272  -3.090  0.00339 **
ventOUEST    -32.272      6.821  -4.731 2.16e-05 ***
ventSUD       -9.507      8.997  -1.057  0.29616
```

L'estimateur de μ, noté ici Intercept, est la moyenne de la concentration en `O3` pour le vent d'`EST`. Les autres valeurs obtenues correspondent aux écarts entre la moyenne de la concentration en `O3` de la cellule pour le vent considéré et la moyenne de la concentration en `O3` pour le vent d'`EST` (cellule de référence).

Les deux colonnes du résumé ci-dessus correspondant au test de nullité d'un paramètre (`t value` et `Pr(>|t|)`) ont un sens pour les 3 dernières lignes du listing. Le test correspond à la question suivante : y a-t-il une ressemblance entre le vent de la cellule de référence (`EST`) et le vent considéré. Selon ces tests, le vent du `SUD` n'est pas différent de celui de l'`EST`, au contraire des vents du `NORD` et d'`OUEST`.

Remarque
Nous pouvons utiliser le contraste « *treatment* », utilisé par défaut en écrivant :

```
> lm(O3 ~ C(vent,treatment), data = ozone)
```

Si nous voulons choisir une cellule témoin spécifique, nous l'indiquons de la manière suivante :

```
> lm(O3 ~ C(vent,base=2), data = ozone)
```

La seconde modalité est choisie comme modalité de référence. Le numéro des modalités correspond à celui des coordonnées du vecteur suivant : `levels(ozone[,"vent"])`.

3. $\sum n_i \alpha_i = 0$. Cette contrainte n'est pas préprogrammée dans R, il faut définir une matrice qui servira de contraste. Cette matrice appelée `contraste` correspond à $X_{[\sum n_i \alpha_i = 0]}$

```
> II <- length(levels(ozone$vent))
> nI <- table(ozone$vent)
> contraste<-matrix(rbind(diag(II-1),
                    -nI[-II]/nI[II]),II,II-1)
```

Et le modèle est donné par l'expression suivante

```
> mod3 <- lm(O3 ~ C(vent,contraste), data = ozone)
```

Nous retrouvons le même tableau d'analyse de la variance que pour `mod2` :

```
> anova(mod3)
Analysis of Variance Table
Response: O3
             Df  Sum Sq Mean Sq F value    Pr(>F)
vent          3  9859.8  3286.6  8.3383 0.0001556 ***
Residuals    46 18131.4   394.2
```

En effet, même si les contraintes changent, les projections $\hat{Y}$ et $\hat{Y}_0$ restent uniques et le test F est identique . L'effet vent semble significatif. Si nous nous intéressons maintenant aux coefficients, nous avons :

```
> summary(mod3)
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)           86.300      2.808  30.737  < 2e-16 ***
C(vent, CONTRASTE)1   17.550      4.093   4.288 9.15e-05 ***
C(vent, CONTRASTE)2   -8.011      5.993  -1.337 0.187858
C(vent, CONTRASTE)3  -14.722      3.744  -3.933 0.000281 ***
```

Nous retrouvons que $\hat{\mu}$ est bien la moyenne de la concentration en `O3`.

4. $\sum \alpha_i = 0$. Cette contrainte est implémentée sous R :

```
> mod4 <- lm(O3 ~ C(vent,sum), data = ozone)
```

Et à nouveau nous retrouvons le même tableau d'analyse de la variance.

```
> anova(mod4)
Analysis of Variance Table
Response: O3
             Df  Sum Sq Mean Sq F value    Pr(>F)
vent          3  9859.8  3286.6  8.3383 0.0001556 ***
Residuals    46 18131.4   394.2
```

L'effet vent est significatif. Si nous nous intéressons maintenant aux coefficients, nous avons :

```
> summary(mod4)
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      87.015      3.027  28.743  < 2e-16 ***
C(vent, sum)1    16.835      4.635   3.632 0.000705 ***
C(vent, sum)2    -8.726      5.573  -1.566 0.124284
C(vent, sum)3   -15.437      4.485  -3.442 0.001240 **
```

Intercept correspond à la moyenne des concentrations moyennes en O3 pour chaque vent.

Enfin il est utile d'analyser les résidus afin de constater si l'hypothèse d'homoscédasticité des résidus est bien vérifiée. Les commandes suivantes permettent d'obtenir des représentations différentes des résidus.

```
> resid2 <- resid(mod2)
> plot(resid2 ~ vent, data=ozone, ylab="residus")
> plot(resid2 ~ jitter(fitted(mod2)),xlab="ychap",ylab="residus")
> xyplot(resid2 ~ I(1:50)|vent, data=ozone,
                          xlab="index", ylab="residus")
```

Nous pouvons constater que (figure 6.10), malgré le faible nombre d'individus par cellule, les variances semblent voisines d'une cellule à l'autre.

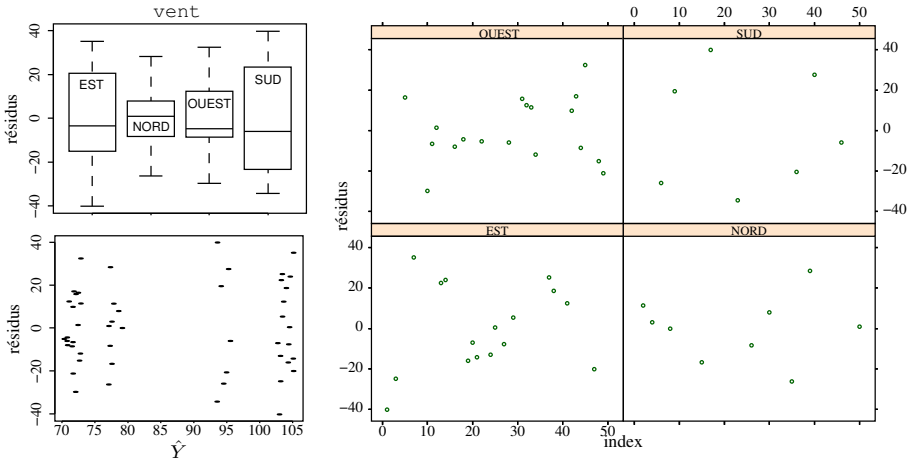


Fig. 6.10 – Trois représentations des résidus.

Nous terminons cette section par une dernière approche de l'analyse de la variance basée sur une décomposition directe de la variance.

6.3.7 Une décomposition directe de la variance

Une introduction très classique de l'analyse de variance consiste à décomposer la variance totale en somme de différentes parties. Rappelons les notations utilisées.

— La variable qualitative explicative admet I modalités (ou niveaux) et le nombre d'individus par niveau vaut n_i. Le nombre total d'individus est $n = \sum_{i=1}^{I} n_i$.

— y_{ij} : observation de la v.a. correspondant à l'individu j du niveau i, où $i = 1, \ldots, I$ et $j = 1, \ldots, n_i$.

— La moyenne empirique par niveau et la moyenne générale sont données par les relations suivantes :

$$\begin{aligned} \bar{y}_i &= y_{i.} = \frac{1}{n_i} \sum_{j=1}^{n_i} y_{ij}, \quad \text{moyenne par niveau } i. \\ \bar{y} &= y_{..} = \frac{1}{n} \sum_{i=1}^{I} \sum_{j=1}^{n_i} y_{ij} = \frac{1}{n} \sum_{i=1}^{I} n_i \bar{y}_i. \end{aligned}$$

L'approche consiste à décomposer la variance totale

$$\frac{1}{n} \sum_{i=1}^{I} \sum_{j=i}^{n_i} (y_{ij} - \bar{y})^2$$

en somme de deux termes. Le premier est une variance intra due au hasard, appelée aussi variance intrastrate (ou résiduelle)

$$\frac{1}{n} \sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2$$

et le second une variance inter due au facteur, appelée aussi variance interstrate (ou des écarts)

$$\frac{1}{n} \sum_{i=1}^{I} n_i (\bar{y}_i - \bar{y})^2.$$

La variance totale étant fixée par les observations, on espère que la partie variance inter soit la plus grande possible, ce qui signifie que les écarts $\bar{y}_i - \bar{y}$ sont grands, donc qu'entre niveaux du facteur l'écart est grand. Plus la variance inter est grande, plus le facteur a de l'importance. On retrouve cette idée dans le test F proposé dans le tableau 6.4 (p. 137) qui est le rapport des variance inter sur la variance résiduelle ramené à leur degré de liberté respectif.

6.4 Analyse de la variance à 2 facteurs

6.4.1 Introduction

Nous voulons maintenant modéliser la concentration en ozone par le vent (4 modalités) et la nébulosité, que nous avons scindée en 2 modalités (soleil-nuageux).

Nous avons mesuré 2 observations par niveau (tableau 6.5).

	NORD	SUD	EST	OUEST
SOLEIL	89.6	134.2	139.0	87.4
	106.6	121.8	126.2	84.0
NUAGEUX	81.2	68.4	63.6	88.0
	78.2	113.8	79.0	41.8

Tableau 6.5 – Concentration en ozone.

En général, la première variable explicative ou premier facteur est celui indiqué en ligne (ici `Nébulosité`) admettant I modalités, la seconde variable explicative ou second facteur est celui indiqué en colonne (ici `Vent`) admettant J modalités. Les individus ne sont plus repérés par un couple (i, j) mais maintenant par un triplet (i, j, k), représentant le k^{e} individu admettant la modalité i de la première variable explicative et la modalité j de la seconde variable explicative. Le nombre n_{ij} correspond au nombre d'observations ayant la modalité i du premier facteur et j du second. Nous avons la définition suivante :

Définition 6.1
Si $\forall(i,j), \quad n_{ij} \geq 1$, *le plan est dit complet.*
Si $\exists(i,j) \ : \ n_{ij} = 0$, *le plan est dit incomplet.*
Si $\forall(i,j), \quad n_{ij} = r$, *le plan est dit équilibré.*

6.4.2 Modélisation du problème

Les deux variables explicatives `Vent` et `Nébulosité` ne sont pas utilisables directement et nous allons donc travailler avec leur version que l'on notera A pour la nébulosité et B pour le vent. Le modèle le plus naturel est

$$y_{ijk} = \mu + \alpha_1 A_{i1} + \alpha_2 A_{i2} + \beta_1 B_{j1} + \beta_2 B_{j2} + \beta_3 B_{j3} + \beta_4 B_{j4} + \varepsilon_{ijk}.$$

Afin d'écrire ce modèle sous forme matricielle, considérons le vecteur $Y \in \mathbb{R}^n$ des observations y_{ijk} rangées dans l'ordre lexicographique de leurs indices. Nous notons $\vec{e}_{ij} \in \mathbb{R}^n$ le vecteur dont toutes les coordonnées sont nulles sauf celles repérées par les indices ijk pour $k = 1, \ldots, n_{ij}$, qui valent 1. Ce vecteur est le vecteur d'appartenance à la cellule (i, j). Les vecteurs $\vec{e}_{ij}$ sont des vecteurs de $\mathbb{R}^n$ orthogonaux entre eux. Nous définissons

$$\vec{e}_{i.} = \sum_j \vec{e}_{ij} \quad \text{et} \quad \vec{e}_{.j} = \sum_i \vec{e}_{ij}$$

où $\vec{e}_{i.}$ est le vecteur d'appartenance à la modalité i du premier facteur et $\vec{e}_{.j}$ est le vecteur d'appartenance à la modalité j du second facteur. Le modèle s'écrit alors

sous la forme suivante

$$Y = \mu\mathbb{1} + \alpha_1\vec{e}_{1.} + \alpha_2\vec{e}_{2.} + \beta_1\vec{e}_{.1} + \beta_2\vec{e}_{.2} + \beta_3\vec{e}_{.3} + \beta_4\vec{e}_{.4} + \varepsilon.$$

ou encore avec les notations précédentes

$$Y = \mu\mathbb{1} + A_c\alpha + B_c\beta + \varepsilon, \tag{6.13}$$

avec $A_c = (\vec{e}_{1.}, \vec{e}_{2.})$ et $B_c = (\vec{e}_{.1}, \vec{e}_{.2}, \vec{e}_{.3}, \vec{e}_{.4})$. Si nous additionnons toutes les colonnes de A_c (idem pour B_c), nous obtenons le vecteur $\mathbb{1}$. La matrice $(\mathbb{1}, A_c, B_c)$ n'est donc pas de plein rang et l'hypothèse $\mathcal{H}_1$ n'est pas vérifiée. Nous ne pouvons donc appliquer directement les résultats des trois chapitres précédents au modèle (6.13). Nous retombons sur les problèmes d'identifiabilité évoqués dans la section précédente et il faudra à nouveau imposer des contraintes.
En régression multiple, nous avons p variables explicatives $X_1, \ldots, X_p$ et nous cherchons le "meilleur" modèle à partir de ces p variables. Nous pouvons bien évidemment considérer des transformations de ces variables ou inclure des interactions (par exemple une nouvelle variable serait $X_1 \times X_2$), comme cela a été indiqué au chapitre 2. En analyse de la variance comme en analyse de la covariance, nous commençons toujours par traiter le modèle avec interaction. Le produit `Nébulosité` avec `Vent` est impossible à effectuer et nous codons ce produit *via* une matrice C_c dont la première colonne indique l'appartenance au croisement `SOLEIL-NORD`, la seconde colonne au croisement `SOLEIL-SUD` et ainsi de suite. Nous obtenons le modèle suivant :

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}, \tag{6.14}$$

avec un effet moyen général μ, un effet différentiel $\alpha_i, i = 1, 2$ pour la nébulosité, un effet différentiel $\beta_j, j = 1, \ldots, 4$ pour le vent et un terme d'interaction γ_{ij}. En utilisant les notations précédentes, l'écriture du modèle sous forme matricielle est :

$$Y = \mu\mathbb{1} + A_c\alpha + B_c\beta + C_c\gamma + \varepsilon,$$

où $C_c = (\vec{e}_{11}, \vec{e}_{12}, \vec{e}_{13}, \vec{e}_{14}, \vec{e}_{21}, \vec{e}_{22}, \vec{e}_{23}, \vec{e}_{24})$. A titre d'exemple, écrivons les matrices obtenues avec le jeu de données présenté :

$$\begin{bmatrix} y_{111} \\ y_{112} \\ y_{121} \\ y_{122} \\ y_{131} \\ y_{132} \\ y_{141} \\ y_{142} \\ y_{211} \\ y_{212} \\ y_{221} \\ y_{222} \\ y_{231} \\ y_{232} \\ y_{241} \\ y_{242} \end{bmatrix} = \begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix} \mu \\ \alpha_1 \\ \alpha_2 \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \gamma_{11} \\ \gamma_{12} \\ \gamma_{13} \\ \gamma_{14} \\ \gamma_{21} \\ \gamma_{22} \\ \gamma_{23} \\ \gamma_{24} \end{bmatrix} + \begin{bmatrix} \varepsilon_{111} \\ \varepsilon_{112} \\ \varepsilon_{121} \\ \varepsilon_{122} \\ \varepsilon_{131} \\ \varepsilon_{132} \\ \varepsilon_{141} \\ \varepsilon_{142} \\ \varepsilon_{211} \\ \varepsilon_{212} \\ \varepsilon_{221} \\ \varepsilon_{222} \\ \varepsilon_{231} \\ \varepsilon_{232} \\ \varepsilon_{241} \\ \varepsilon_{242} \end{bmatrix}$$

Remarquons que les interactions de variables continues, construites avec le produit des variables, et l'interaction de 2 facteurs, représentée ici par C, suivent la même logique de construction. En effet les colonnes de C_c sont tout simplement le résultat des produits 2 à 2 des colonnes de A_c par celles de B_c.
A nouveau la matrice $(\mathbb{1}, A, B, C)$ n'est pas de plein rang et l'hypothèse $\mathcal{H}_1$ n'est pas vérifiée. La matrice $X = (\mathbb{1}, A, B, C)$ de taille $n \times (1 + I + J + IJ)$ est de rang IJ. Il faut imposer donc $1 + I + J$ contraintes linéairement indépendantes afin qu'elle devienne inversible. Les contraintes classiques sont :

1. Contrainte de type analyse par cellule

$$\mu = 0, \qquad \forall i \quad \alpha_i = 0, \qquad \forall j \quad \beta_j = 0.$$

2. Contrainte de type cellule de référence

$$\alpha_1 = 0, \qquad \beta_1 = 0, \qquad \forall i \quad \gamma_{i1} = 0, \qquad \forall j \quad \gamma_{1j} = 0.$$

3. Contrainte de type somme

$$\sum_i \alpha_i = 0, \qquad \sum_j \beta_j = 0, \qquad \forall i \quad \sum_j \gamma_{ij} = 0, \qquad \forall j \quad \sum_i \gamma_{ij} = 0.$$

Remarque
Pour les contraintes de type analyse par cellule ou cellule de référence, nous avons bien $1 + 1 + I + (J - 1)$ contraintes. En effet, la dernière contrainte $\gamma_{1j} = 0$ pour $j = 1, \cdots, J$ pourrait s'écrire $\gamma_{1j} = 0$ pour j variant de 2 à J. Le cas correspondant à $j = 1$, soit γ_{11}, est déjà donné dans la contrainte précédente.
Pour la contrainte de type somme, c'est plus difficile à voir. Montrons que les $I + J$ contraintes $\forall i \sum_j \gamma_{ij} = 0$ et $\forall j \sum_i \gamma_{ij} = 0$ ne sont pas indépendantes. En effet quand $I + J - 1$ contraintes sont vérifiées, la dernière restante l'est aussi.

$$\begin{array}{|ccccc|c}
\hline
c_{11} & c_{12} & \dots & c_{1J-1} & c_{1J} & = 0 \\
c_{21} & c_{22} & \dots & c_{2J-1} & c_{2J} & = 0 \\
\vdots & \vdots & & \vdots & \vdots & \vdots \\
c_{I1} & c_{I2} & \dots & c_{IJ-1} & c_{IJ} & = 0 \\
\hline
= 0 & = 0 & \dots & = 0 & c &
\end{array}$$

Posons que $I + J - 1$ contraintes sont vérifiées : I en ligne et $J - 1$ en colonnes (voir ci-dessus). La dernière somme c vaut 0 (voir ci-dessus).

6.4.3 Estimation des paramètres

Nous n'aborderons ici que la contrainte de type analyse par cellule et la contrainte de type somme et *nous considérerons uniquement les plans équilibrés avec r observations par cellule.*

Considérons les notations suivantes :

$$\bar{y}_{ij} = \frac{1}{r}\sum_{k=1}^{r} y_{ijk}, \quad \bar{y}_{i.} = \frac{1}{Jr}\sum_{j=1}^{J}\sum_{k=1}^{r} y_{ijk}, \quad \bar{y}_{.j} = \frac{1}{Ir}\sum_{i=1}^{I}\sum_{k=1}^{r} y_{ijk}, \quad \bar{y} = \frac{1}{n}\sum_{i,j,k} y_{ijk}.$$

Proposition 6.3
Soit le modèle d'analyse de la variance à 2 facteurs suivant :

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}.$$

1. Sous les contraintes $\mu = 0$, $\alpha_i = 0$ pour tout $i = 1, \ldots, I$ et $\beta_j = 0$ pour tout $j = 1, \ldots, J$, qui correspond au modèle $y_{ijk} = \gamma_{ij} + \varepsilon_{ijk}$, les estimateurs des moindres carrés des paramètres inconnus sont

$$\hat{\gamma}_{ij} = \bar{y}_{ij}$$

Les $\hat{\gamma}_{ij}$ correspondent aux moyennes par cellule.

2. Sous les contraintes $\sum_i \alpha_i = 0$, $\sum_j \beta_j = 0$, $\forall i \ \sum_j \gamma_{ij} = 0$ et $\forall j \ \sum_i \gamma_{ij} = 0$, les estimateurs des moindres carrés des paramètres inconnus sont

$$\begin{aligned}
\hat{\mu} &= \bar{y} \\
\hat{\alpha}_i &= \bar{y}_{i.} - \bar{y} \\
\hat{\beta}_j &= \bar{y}_{.j} - \bar{y} \\
\hat{\gamma}_{ij} &= \bar{y}_{ij} - \bar{y}_{i.} - \bar{y}_{.j} + \bar{y}.
\end{aligned}$$

Dans tous les cas, la variance résiduelle σ^2 est estimée par

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{r}(y_{ijk} - \hat{y}_{ij})^2}{n - IJ}.$$

La preuve est à faire en exercice (voir exercices 6.4 et 6.5).

6.4.4 Analyse graphique de l'interaction

Une des principales questions de l'ANOVA à deux facteurs est de savoir si les facteurs ont une influence sur la variable à expliquer. La première analyse à effectuer consiste à étudier l'interaction. En effet, si l'interaction a un sens, alors les facteurs A et B influent sur la variable à expliquer car l'interaction est le produit de A avec B. Considérons le modèle complet

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk},$$

que nous pouvons réécrire sous une forme simplifiée

$$y_{ijk} = m_{ij} + \varepsilon_{ijk}.$$

Considérons maintenant le modèle sans interaction

$$y_{ijk} \quad = \quad \mu + \alpha_i + \beta_j + \varepsilon_{ijk}.$$

La première étape consiste à tester la significativité de l'interaction. Cela revient à tester le modèle avec interaction contre le modèle sans l'interaction. Avant d'aborder les tests, nous étudions une approche graphique de l'interaction.
Fixons le facteur A au niveau i. Pour ce niveau donné, nous avons J cellules, chacune correspondant à un niveau du facteur B. Prenons l'espérance dans chacune de ces cellules, nous obtenons sous l'hypothèse que l'interaction n'est pas significative : $\mu + \alpha_i + \beta_j$, $1 \leq j \leq J$. En traçant en abscisse le numéro j de la cellule et en ordonnée son espérance, nous obtenons une ligne brisée appelée *profil*.
Passons au niveau α_{i+1} du facteur A. Nous pouvons tracer la même ligne brisée et ce profil sera, sous l'hypothèse de non-interaction : $\mu + \alpha_{i+1} + \beta_j$, soit le profil précédent translaté verticalement de $\alpha_{i+1} - \alpha_i$.
Une absence d'interaction se traduit donc par des profils parallèles. Un moyen de visualiser l'interaction est donc de tracer les profils estimés sous le modèle complet (avec interaction) et de regarder si les lignes brisées sont parallèles. Sur l'exemple de l'ozone, nous obtenons grâce aux ordres suivants :

```
> par(mfrow=c(1,2))
> with(ozone, interaction.plot(vent, nebulosite, O3, col=1:2))
> with(ozone, interaction.plot(nebulosite, vent, O3, col=1:4))
```

les profils suivants :

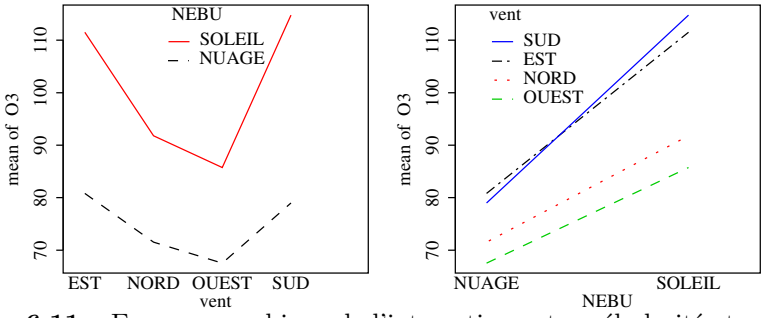


Fig. 6.11 – Examen graphique de l'interaction entre nébulosité et vent.

Les profils ne sont pas exactement parallèles mais quasiment. Nous constatons que les modalités `EST-SOLEIL` et `SUD-SOLEIL` sont un peu éloignées de la position qu'elles auraient dû occuper si les profils étaient exactement parallèles. Le vent de SUD associé à un temps ensoleillé semble très légèrement plus propice à un fort pic d'ozone. Ces graphiques suggèrent une très légère interaction entre `Vent` et `Nébulosité`, principalement entre `SUD` et `SOLEIL`. Mais est-ce que cette différence locale est suffisante par rapport aux différences entre individus dues à la variabilité

ε ? Afin de répondre à cette question il est nécessaire d'utiliser un test statistique et de supposer l'hypothèse gaussienne vérifiée.

6.4.5 Hypothèse gaussienne et test de l'interaction

Grâce à l'hypothèse gaussienne sur les erreurs ε, nous pouvons utiliser les tests d'hypothèses vus au chapitre 5. Rappelons encore que notre principal objectif est de *savoir si les facteurs influent sur la variable à expliquer.*
Nous préconisons de tester en premier la significativité de l'interaction. En effet, si l'interaction est significative, les 2 facteurs sont influents *via* leur interaction, il n'est donc pas nécessaire de tester leur influence respective.
On se place dans le modèle avec interaction

$$y_{ijk} \quad = \quad \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$$

dans lequel on souhaite tester

$$(\mathrm{H}_0)_{AB} : \quad \forall (i,j) \quad \gamma_{ij} = 0 \qquad \text{contre} \qquad (\mathrm{H}_1)_{AB} : \exists (i,j) \quad \gamma_{ij} \neq 0.$$

Les modèles sous $(\mathrm{H}_0)_{AB}$ et $(\mathrm{H}_1)_{AB}$ peuvent s'écrire encore sous la forme suivante :

$$\begin{aligned} y_{ijk} &= \mu + \alpha_i + \beta_j + \varepsilon_{ijk}, \quad \text{modèle sous} \quad (\mathrm{H}_0)_{AB} \\ y_{ijk} &= \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}, \quad \text{modèle sous} \quad (\mathrm{H}_1)_{AB}. \end{aligned}$$

Ce test, qui permet de connaître l'influence globale de l'interaction des facteurs, est tout simplement un test entre deux modèles dont l'un est un cas particulier de l'autre (section 5.5.2, p. 98). Nous pouvons donc énoncer le théorème suivant.

Théorème 6.2
Soit un modèle d'analyse de la variance à 2 facteurs A et B. Notons l'hypothèse nulle (modèle restreint) $(\mathrm{H}_0)_{AB}$: $\forall (i,j) \quad \gamma_{ij} = 0$, qui correspond au modèle $y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}$, contre l'hypothèse alternative $(\mathrm{H}_1)_{AB} : \exists (i,j) \quad \gamma_{ij} \neq 0$ qui correspond au modèle complet $y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$. Pour tester ces deux hypothèses, nous utilisons la statistique de test F ci-dessous qui possède comme loi sous $(\mathrm{H}_0)_{AB}$:

$$F = \frac{\|\hat{Y} - \hat{Y}_0\|^2/(IJ - I - J + 1)}{\|Y - \hat{Y}\|^2/(n - IJ)} \sim \mathcal{F}_{IJ-I-J+1,n-IJ}.$$

Lorsque le plan est équilibré, la statistique de test s'écrit :

$$F = \frac{r \sum_i \sum_j (\bar{y}_{ij} - \bar{y}_{i.} - \bar{y}_{.j} + \bar{y})^2}{\sum_i \sum_j \sum_k (y_{ijk} - \bar{y}_{ij})^2} \frac{n - IJ}{I + J - 1} \sim \mathcal{F}_{IJ-I-J+1,n-IJ}.$$

L'hypothèse $(\mathrm{H}_0)_{AB}$ sera rejetée en faveur de $(\mathrm{H}_1)_{AB}$ au niveau α si l'observation de la statistique F est supérieure à $f_{IJ-I-J+1,n-IJ}(1-\alpha)$, et nous conclurons alors à l'effet des facteurs explicatifs.

La preuve de ce théorème se fait facilement. Il suffit d'appliquer le théorème 5.2 p. 100 avec l'écriture des normes données en (6.11) et (6.12). Nous avons un premier modèle, ou modèle complet,

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk} \quad \text{modèle (1)}$$

et obtenons les estimations suivantes : $\hat{\mu}(1), \cdots, \hat{Y}(1)$ et $\hat{\sigma}^2(1)$, le (1) précise que nous sommes dans le premier modèle.

Si l'interaction est significative, nous conservons le modèle (1). Sinon, nous le rejetons au profit du modèle (2)

$$y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk} \quad \text{modèle (2)}$$

dans lequel nous obtenons les estimations $\hat{\mu}(2), \cdots, \hat{Y}(2)$ et $\hat{\sigma}^2(2)$.

L'étape suivante consiste à tester l'influence des facteurs A et/ou B et donc tenter de simplifier le modèle. Testons par exemple l'influence du facteur A. Nous avons déjà le modèle (2) qui prend en compte l'effet de A, ce qui sera donc l'hypothèse alternative $(\mathrm{H}_1)_A$. En simplifiant ce modèle pour éliminer l'influence de A nous obtenons le modèle (3) qui sera l'hypothèse nulle du test, $(\mathrm{H}_0)_A$,

$$y_{ijk} = \mu + \beta_j + \varepsilon_{ijk} \quad \text{modèle (3)}$$

avec les estimations suivantes : $\hat{\mu}(3), \cdots, \hat{Y}(3)$ et $\hat{\sigma}^2(3)$.
Pour tester l'influence du facteur A, nous cherchons à départager 2 modèles, le modèle (2) et le modèle (3) et nous avons la statistique de test

$$F = \frac{\|\hat{Y}(2) - \hat{Y}(3)\|^2/(I-1)}{\hat{\sigma}^2} \sim \mathcal{F}_{(I-1),ddl(\text{résiduelle})},$$

où $ddl(\text{résiduelle})$ représente la dimension de l'espace dans lequel on projette Y pour obtenir l'estimation de σ^2.

1. Si nous sommes dans la logique des tests entre modèles emboîtés, le premier modèle a été rejeté, nous travaillons donc avec les modèles (2) et (3), nous estimons alors σ^2 par $\hat{\sigma}^2(2)$. La statistique de test vaut

$$F = \frac{\|\hat{Y}(2) - \hat{Y}(3)\|^2/(I-1)}{\|Y - \hat{Y}(2)\|^2/(n-I-J+1)} \sim \mathcal{F}_{(I-1),(n-I-J+1)}.$$

2. Bien que l'on ait rejeté le modèle complet avec interaction, certains auteurs et utilisateurs préconisent de conserver le modèle complet pour estimer σ^2 en arguant de la précision de cet estimateur. Il est vrai que la SCR obtenue dans le modèle complet est plus petite que la SCR obtenue dans le modèle sans interaction, mais les degrés de liberté associés sont différents. Ainsi, dans le modèle complet, le ddl vaut $n - IJ$ alors que dans le modèle sans interaction, le ddl vaut $n - I - J + 1$.

La précision accrue de l'estimateur peut être vue comme une précaution envers la possibilité d'une interaction, même si on l'a rejetée par le test d'hypothèse $(\mathrm{H}_0)_{AB}$ contre $(\mathrm{H}_1)_{AB}$. Dans ce cas, la statistique de test vaut

$$F = \frac{\|\hat{Y}(2) - \hat{Y}(3)\|^2/(I-1)}{\|Y - \hat{Y}(1)\|^2/(n-IJ)} \sim \mathcal{F}_{(I-1),(n-IJ)}.$$

En pratique et de façon historique, les résultats d'une analyse de la variance sont présentés dans un tableau récapitulatif, appelé tableau d'analyse de la variance. Ce tableau est difficile à analyser lorsque le plan d'expérience n'est pas équilibré (*i.e.* même nombre de répétitions pour chaque niveau i et j des facteurs). Dans le cas d'un plan équilibré, les calculs sont simplifiés et les interprétations également. Nous renvoyons le lecteur à l'exercice 6.6.

Conclusion

Résumons donc la mise en œuvre d'une analyse de la variance à deux facteurs. Il est utile de commencer par examiner graphiquement l'interaction. Ensuite nous pouvons toujours supposer l'hypothèse gaussienne vérifiée et commencer par tester l'hypothèse d'interaction $(\mathrm{H}_0)_{AB}$. Comme le test dépend de projections qui sont uniques, il est inchangé quel que soit le type de contrainte utilisé. Ensuite, si l'interaction n'est pas significative, il est possible de tester les effets principaux $(\mathrm{H}_0)_A$ et $(\mathrm{H}_0)_B$ et de conclure. Enfin l'analyse des résidus permet quant à elle de confirmer l'hypothèse d'homoscédasticité et l'hypothèse de normalité.

Pour une présentation plus complète de l'analyse de la variance nous renvoyons le lecteur intéressé au livre de Scheffé (1959). De même un traitement complet des plans d'expérience peut être trouvé dans Droesbeke *et al.* (1997).

6.4.6 Exemple : la concentration en ozone

Afin de savoir si les variables `Vent` et `Nébulosité` ont un effet sur la concentration d'ozone, nous allons utiliser une ANOVA à deux facteurs. N'ayant aucune autre connaissance *a priori*, tous les modèles incluant le vent sont possibles : avec interaction, sans interaction, sans effet du facteur `Nébulosité`. Il est conseillé de commencer par le modèle avec le plus d'interaction et ensuite d'essayer d'éliminer les interactions. Ainsi nous pouvons tester $(\mathrm{H}_0)_{AB}, \quad y_{ijk} = \alpha_i + \beta_j + \varepsilon_{ijk}\ \forall(i,j,k)$ contre $(\mathrm{H}_1)_{AB}, \quad y_{ijk} = \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}\ \forall(i,j,k)$. Ces deux modèles s'écrivent et se testent sous R de la façon suivante :

```
> mod1 <- lm(O3 ~ vent + nebulosite + vent:nebulosite, data=ozone)
> mod2 <- lm(O3 ~ vent + nebulosite, data = ozone)
```

```
> anova(mod2, mod1)
Analysis of Variance Table
Model 1: O3 ~ vent + nebulosite
Model 2: O3 ~ vent + nebulosite + vent:nebulosite
  Res.Df   RSS Df Sum of Sq     F Pr(>F)
1     45 11730
2     42 11246  3    483.62 0.602 0.6173
```

L'hypothèse de non-interaction $(\mathrm{H}_0)_{AB}$ est donc conservée. La différence constatée graphiquement (fig. 6.11) n'est pas suffisante pour repousser l'hypothèse de non interaction.

Nous souhaitons savoir si la nébulosité possède un effet sur la concentration en ozone. Nous testons alors $(\mathrm{H}_0)_B, \quad y_{ijk} = \mu + \alpha_i + \varepsilon_{ijk}\ \forall (i,j,k)$ contre $(\mathrm{H}_1)_B, \quad y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}\ \forall (i,j,k)$. Nous allons donc utiliser la statistique F_B mais avec quel estimateur $\hat{\sigma}^2$? Nous avons deux choix (voir p. 149)

— Le premier consiste à utiliser $\|Y - \hat{\mu} - \hat{\alpha}_i - \hat{\beta}_j\|^2/(n - I - J + 1)$, qui est l'estimateur classique de $\hat{\sigma}^2$ dans un test entre modèles emboîtés.
— Le second consiste à conserver l'estimateur de σ^2 utilisé lors du test précédent $(\mathrm{H}_0)_{AB}$ (test d'existence d'interaction) où l'estimateur était : $\|Y - \hat{\mu} - \hat{\alpha}_i - \hat{\beta}_j - \hat{\gamma}_{ij}\|^2/(n - IJ)$.

La première méthode consiste à dire, puisque le modèle sans interaction a été conservé, qu'il est donc « vrai » et on l'utilise pour estimer l'erreur. La seconde méthode consiste à dire, bien que le modèle à interaction ait été repoussé, il se peut qu'il subsiste une interaction même faible qui pourrait modifier l'estimation de σ^2. Afin d'éviter cette modification, la même estimation de σ^2 est conservée. Pour réaliser cela, nous introduisons un nouveau modèle sans effet nébulosité que nous testons ensuite selon la première procédure :

```
> mod3 <- lm(O3 ~ vent, data = ozone)
> anova(mod3, mod2)
Model 1: O3 ~ vent
Model 2: O3 ~ vent + nebulosite
  Res.Df   RSS Df Sum of Sq      F    Pr(>F)
1     46 18131
2     45 11730  1    6401.5 24.558 1.066e-05 ***
```

et nous repoussons $(\mathrm{H}_0)_B$, il existe un effet du vent et de la nébulosité. Si l'on utilise la première procédure, nous pouvons résumer l'ensemble des tests emboîtés réalisés jusqu'ici avec :

```
> anova(mod3, mod2, mod1)
Model 1: O3 ~ vent
Model 2: O3 ~ vent + nebulosite
Model 3: O3 ~ vent + nebulosite + vent:nebulosite
  Res.Df   RSS Df Sum of Sq      F    Pr(>F)
1     46 18131
2     45 11730  1    6401.5 23.907 1.523e-05 ***
3     42 11246  3     483.6  0.602    0.6173
```

nous lisons encore une fois qu'au niveau de 5 % l'hypothèse $(\mathrm{H}_0)_B$ est rejetée (voir ligne 2). L'analyse des résidus ne donne rien de particulier ici et est donc omise. Si vous souhaitez toutefois retrouver le tableau 6.6, il suffit d'utiliser l'ordre suivant

```
> anova(mod1)
```

mais nous ne recommandons pas son usage ici puisque le plan n'est pas équilibré et l'ordre des facteurs influe sur la décomposition.

6.5 Exercices

Exercice 6.1 (Questions de cours)

1) Vous faites une analyse de la variance à 1 facteur équilibrée, la variance de l'estimateur des MC est diagonale :
   - A oui, toujours,
   - B non, jamais,
   - C peut-être, cela dépend des données de X.

2) Lors d'une analyse de la variance à 2 facteurs, le modèle utilisé est $y_{ijk} = m_{ij} + \varepsilon_{ijk}$. Les paramètres estimés sont $\hat{m}_{ij}$, la région de confiance de deux paramètres est :
   - A une ellipse dont les axes sont parallèles aux axes du repère,
   - B une ellipse dont les axes peuvent ne pas être parallèles aux axes du repère,
   - C un cercle.

3) Lors d'une analyse de la variance à 2 facteurs, le modèle utilisé est $y_{ijk} = m_{ij} + \varepsilon_{ijk}$ et le plan équilibré. Les paramètres estimés sont $\hat{m}_{ij}$, la région de confiance de deux paramètres est :
   - A une ellipse dont les axes sont parallèles aux axes du repère,
   - B une ellipse dont les axes peuvent ne pas être parallèles aux axes du repère,
   - C un cercle.

4) Vous souhaitez tester l'effet d'un facteur lors d'une analyse de la variance à 2 facteurs, l'interaction est positive
   - A vous effectuez l'analyse à un facteur correspondant et concluez en conséquence,
   - B vous ne faites rien car il y a un effet du facteur,
   - C vous regardez dans le tableau de l'ANOVA la valeur de la p-value de l'effet désiré afin de conclure.

Exercice 6.2 (Analyse de la covariance)

Nous souhaitons expliquer une variable Y par une variable continue et une variable qualitative admettant I modalités.

1) Donner la forme explicite des matrices X pour les 3 modélisations proposées.

2) Calculer ensuite l'estimateur des MC obtenu dans le modèle 6.1.
3) Montrer que cet estimateur peut être obtenu en effectuant I régressions simples.

Exercice 6.3 (†Estimateurs des MC en ANOVA à 1 facteur)
Démontrer la proposition 6.2 p. 135.

Exercice 6.4 (Estimateurs des MC en ANOVA à deux facteurs)
Démontrer la proposition 6.3 p. 146 lorsque les contraintes sont de type analyse par cellule.

Exercice 6.5 (††Estimateurs des MC en ANOVA à deux facteurs suite)
Démontrer la proposition 6.3 p. 146 lorsque les contraintes sont de type somme dans un plan équilibré.

Exercice 6.6 (†Tableau d'ANOVA à 2 facteurs équilibrée)
Considérons un plan équilibré et notons le vecteur $\vec{Y} \in \mathbb{R}^{IJr}$ des observations Y_{ijk} rangées dans l'ordre lexicographique de leurs indices (i facteur A et j facteur B) soit

$$\vec{Y}^t = (Y_{111}, Y_{112}, \ldots, Y_{11n}, Y_{211}, Y_{212}, \ldots, Y_{pqn}).$$

Nous notons $\vec{e}_{ijk}$ de $\mathbb{R}^n$ où $n = IJr$ le vecteur dont toutes coordonnées sont nulles sauf celle indicée par ijk et $\vec{e}_{ij}$ le vecteur dont toutes les coordonnées sont nulles sauf celles repérées par les indices ijk pour $k = 1, \ldots, n$, les coordonnées non nulles valant 1. On définit

$$\vec{e} := \sum_{ij} \vec{e}_{ij} \qquad \vec{e}_{i.} := \sum_{j} \vec{e}_{ij} \qquad \vec{e}_{.j} := \sum_{j} \vec{e}_{ij}.$$

Nous définissons les valeurs suivantes :

$$Y_{...} = \frac{1}{IJr} \sum_{ijk} Y_{ijk} \quad Y_{i..} = \frac{1}{Jr} \sum_{jk} Y_{ijk} \quad Y_{.j.} = \frac{1}{Ir} \sum_{ik} Y_{ijk} \quad Y_{ij.} = \frac{1}{r} \sum_{k} Y_{ijk}.$$

Nous souhaitons estimer les paramètres du modèle suivant :

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}. \tag{6.15}$$

Ce modèle est sur-paramétré, nous devons imposer des contraintes identifiantes. Nous souhaitons travailler avec les contraintes suivantes $\sum_i \alpha_i = 0$, $\sum_j \beta_j = 0$, pour tout i $\sum_j (\alpha\beta)_{ij} = 0$ et pour tout j $\sum_i (\alpha\beta)_{ij} = 0$. Ces contraintes sont appelées contraintes d'orthogonalité. On définit les sous-espaces suivants :

$$E_1 := \{\mu\vec{e},\ \mu \text{ quelconque}\} \qquad E_2 := \{\sum_i a_i \vec{e}_{i.},\ \sum_i a_i = 0\}$$

$$E_3 := \{\sum_j b_j \vec{e}_{.j},\ \sum_j b_j = 0\} \qquad E_4 := \{\sum_{ij} c_{ij} \vec{e}_{ij},\ \sum_i c_{ij} = 0 \text{ et } \sum_j c_{ij} = 0\}.$$

1) Montrez que le modèle de l'analyse de la variance s'écrit alors

$$\vec{Y} = \mu\vec{e} + \sum_i \alpha_i \vec{e}_{i.} + \sum_j \beta_j \vec{e}_{.j} + \sum_{ij} (\alpha\beta)_{ij} \vec{e}_{ij} + \vec{\varepsilon}.$$

2) Montrez que les sous-espaces E_i sont deux à deux orthogonaux. On note alors

$$E := E_1 \overset{\perp}{\bigoplus} E_2 \overset{\perp}{\bigoplus} E_3 \overset{\perp}{\bigoplus} E_4.$$

3) Montrez que les dimensions de E_1, E_2, E_3, E_4 sont respectivement 1, $I-1$, $J-1$, $(I-1)(J-1)$.
4) On note Q, P, P_1, P_2, P_3 et P_4 les projections orthogonales sur les sous-espaces $E^{\perp}$, E, E_1, E_2, E_3 et E_4. Calculez

$$P_1(\vec{Y}) \quad P_2(\vec{Y}) \quad P_3(\vec{Y}) \quad P_4(\vec{Y}) \quad Q(\vec{Y})$$

en fonction des $Y_{...}, Y_{i..}, Y_{.j.}, Y_{ij.}$.
5) Estimer les paramètres du modèle sous ces contraintes.
6) En déduire l'équation de l'analyse de la variance.
7) Montrez que

$$\frac{||P_i(\vec{Y})||^2}{\sigma^2} \sim \chi^2(dim(E_i), ||P_i(\vec{m})||^2)$$

où $\vec{m}$ est le vecteur des moyennes de $\vec{Y}$.
8) Déduisez de la question précédente que SCE_a, SCE_b, $SCE_{(ab)}$ et SCR sont indépendantes. Donnez leur loi respective.
9) Nous notons $CME_{(ab)}$ et CMR les carrés moyens de l'interaction et résiduel. Montrez que le rapport $CME_{(ab)}/CMR$ suit une loi de Fisher décentrée dont vous donnerez les degrés de liberté et le décentrage.
10) Vous pouvez donc maintenant analyser le tableau d'analyse de la variance donné ci-dessous.

Variation	ddl	SC	CM	valeur du F	$\Pr(> F)$
facteur A	I-1	SC_A	$\mathrm{CM}_A = \dfrac{\mathrm{SC}_A}{(I-1)}$	$\dfrac{\mathrm{CM}_A}{\mathrm{CMR}}$	
facteur B	J-1	SC_B	$\mathrm{CM}_B = \dfrac{\mathrm{SC}_B}{(J-1)}$	$\dfrac{\mathrm{CM}_B}{\mathrm{CMR}}$	
Interaction	(I-1)(J-1)	SC_{AB}	$\mathrm{CM}_{AB} = \dfrac{\mathrm{SC}_{AB}}{(I-1)(J-1)}$	$\dfrac{\mathrm{CM}_{AB}}{\mathrm{CMR}}$	
Résiduelle	n-IJ	SCR	$\mathrm{CMR} = \dfrac{\mathrm{SCR}}{(n-IJ)}$		

Tableau 6.6 – Tableau d'analyse de la variance.

La première colonne indique la source de la variation, puis le degré de liberté associé à chaque effet. La somme des carrés (SCR) est donnée avant le carré moyen (CM), qui est par définition la SCR divisée par le ddl. Ainsi, dans le cas où les sous-espaces E_1, E_2, E_3 et E_4 sont orthogonaux, ce tableau donne tous les tests indiqués précédemment, en utilisant l'estimation de σ^2 donnée par le modèle avec interaction.

— La statistique de test d'interaction, $(\mathrm{H}_0)_{AB}$ contre $(\mathrm{H}_1)_{AB}$, est CM_{AB} / CMR .
— La statistique de test d'influence du facteur A, $(\mathrm{H}_0)_A$ contre $(\mathrm{H}_1)_A = (\mathrm{H}_0)_{AB}$, est CM_A / CMR .
— La statistique de test d'influence du facteur B, $(\mathrm{H}_0)_B$ contre $(\mathrm{H}_1)_B = (\mathrm{H}_0)_{AB}$, est CM_B / CMR .

Ce tableau d'analyse de variance est donc une présentation synthétique des tests d'influence des différents facteurs et interaction.

6.6 Note : identifiabilité et contrastes

Nous avons $X = (\mathbb{1}, A_c)$ où A_c, de taille $n \times I$ est de rang I. La matrice X de taille $n \times p$ ($p = I + 1$) n'est pas de plein rang mais de rang I et $\dim(\Im(X)) = I$ et non pas $I + 1$. Rappelons que la matrice X peut être vue comme la matrice dans les bases canoniques d'une application linéaire f de $\mathbb{R}^p$ dans $\mathbb{R}^n$. En identifiant X et f ainsi que les vecteurs de $\mathbb{R}^p$ (et $\mathbb{R}^n$) à leurs coordonnées dans la base canonique de $\mathbb{R}^p$ (et $\mathbb{R}^n$), nous avons

$$\begin{aligned} X \quad : \quad & \mathbb{R}^p \to \mathbb{R}^n \\ & \beta \mapsto X(\beta) = X\beta. \end{aligned}$$

L'espace de départ $\mathbb{R}^p$ est l'espace des coefficients, l'espace d'arrivée $\mathbb{R}^n$ celui des variables. Ces espaces sont munis d'un produit scalaire, le produit scalaire euclidien. On peut décomposer chacun de ces 2 espaces en 2 espaces supplémentaires orthogonaux. Nous cherchons un vecteur de coefficients, élément de $\mathbb{R}^p$ qui se décompose en :

$$\mathbb{R}^p \quad = \quad \ker(X) \oplus \ker(X)^{\perp},$$

avec $\ker(X) = \{\beta \in \mathbb{R}^p \ : \ X\beta = 0\}$ le noyau de X. Donc pour un coefficient quelconque $\gamma \in \mathbb{R}^p$, nous pouvons l'écrire comme

$$\gamma \quad = \quad \gamma^{\dagger} + \gamma^{\ddagger}, \ \ \gamma^{\dagger} \in \ker(X) \text{ et } \gamma^{\ddagger} \in \ker(X)^{\perp}.$$

Si on prend maintenant un coefficient $\hat{\beta}$ qui minimise les MC, nous avons

$$\hat{\beta} \quad = \quad \hat{\beta}^{\dagger} + \hat{\beta}^{\ddagger}, \text{ avec } X\hat{\beta} = X\hat{\beta}^{\dagger} + X\hat{\beta}^{\ddagger} = X\hat{\beta}^{\ddagger}.$$

En ajoutant à $\hat{\beta}^{\ddagger}$ n'importe quel élément $\beta^{\dagger}$ de $\ker(X)$, on a toujours $\hat{\beta}^{\dagger} + \beta^{\ddagger}$ solution des MC. Il n'y a pas unicité. Si l'on souhaite un unique vecteur de coefficient solution des MC, il semble naturel de poser que $\beta^{\dagger} = 0$ et de garder $\hat{\beta}^{\ddagger} \in \ker(X)^{\perp}$ comme solution du problème. Nous cherchons donc l'élément (unique) $\hat{\beta}^{\ddagger} \in \ker(X)^{\perp}$ solution des MC.

Solution de norme minimum

Montrons que le vecteur $\hat{\beta}^{\ddagger}$, qui est le vecteur solution du problème et qui est élément de $\ker(X)^{\perp}$, est le vecteur solution des MC qui est de norme minimum.
Soit un vecteur quelconque $\hat{\beta}$ solution des MC, il se décompose en 2 parties orthogonales, et du fait de cette orthogonalité nous avons la décomposition suivante

$$\|\hat{\beta}\|^2 \quad = \quad \|\hat{\beta}^{\dagger} + \hat{\beta}^{\ddagger}\|^2 = \|\hat{\beta}^{\dagger}\|^2 + \|\hat{\beta}^{\ddagger}\|^2 \geq \|\hat{\beta}^{\ddagger}\|^2.$$

Nous avons donc que $\hat{\beta}^{\ddagger}$ est la solution des MC de norme minimum.
Une première approche donne directement $\hat{\beta}^{\ddagger} = (X'X)^{+}X'Y$, où $(X'X)^{+}$ est l'inverse généralisé de Moore-Penrose (voir Golub & Van Loan, 1996, pp. 256-257).
Une autre approche consiste à utiliser une solution du problème des MC quelconque et de la projeter dans $\ker(X)^{\perp}$. Pour cela, il nous faut déterminer $\ker(X)^{\perp}$, ou plus simplement $\ker(X)$. Quelle est la dimension de $\ker(X)$?
Rappelons le théorème du rang :

$$\dim(\Im(X)) + \dim(\ker(X)) = p = I + 1,$$

où p est la dimension de l'espace de départ de l'application linéaire associée à X (ou le nombre de colonne de X). Ici nous savons que $\dim(\Im(X)) = I$ et donc $\dim(\ker(X)) = 1$.

Le sous-espace vectoriel $\ker(X)$ est engendré par 1 vecteur non nul de $\mathbb{R}^p$, vecteur que nous pouvons noter $\beta^\dagger$. Nous savons donc que $\ker(X)^\perp$ est engendré par $I = p-1$ vecteurs. En termes de coefficients, cela se traduit par la phrase suivante : si l'on souhaite avoir un vecteur de coefficients unique, on ne pourra avoir que $p-1$ coefficients indépendants, le dernier se déduira des autres par une combinaison linéaire.
Trouvons maintenant un vecteur $\beta^\dagger$ non nul de $\ker(X)$, formant ainsi une base de $\ker(X)$. Si nous posons que $\beta^\dagger = (-1, 1, \ldots, 1)'$, il est bien sûr non nul. Nous savons que $X = (\mathbb{1}, A_c)$ mais aussi que la somme des colonnes de A_c vaut $\mathbb{1}$, donc lorsque l'on effectue $X\beta^\dagger$ nous trouvons O_n et donc $\beta^\dagger = (-1, 1, \ldots, 1)'$ est une base de $\ker(X)$. Tout vecteur orthogonal à $\beta^\dagger$ sera dans $\ker(X)^\perp$, et il suffit donc de projeter une solution $\hat{\beta}$ des MC dans l'orthogonal de $\beta^\dagger$ pour obtenir la solution de norme minimum $\beta^\ddagger$:

$$\beta^\ddagger \quad = \quad (I_n - \beta^\dagger(\beta^{\dagger\prime}\beta^\dagger)^{-1}\beta^{\dagger\prime})\hat{\beta}.$$

Cette solution offre l'intérêt d'être la plus faible en norme, cependant elle n'est pas forcément interprétable au niveau des coefficients, dans le sens où l'on ne contrôle pas la contrainte linéaire reliant les coefficients entre eux.

Contrastes

Une autre approche combine l'élégance de la solution de norme minimum (pas de choix arbitraire) à l'interprétabilité. Cette approche part du constat que souvent, le praticien n'est pas intéressé par les coefficients en soi mais par leur différence ou toute autre combinaison linéaire des coefficients. Par exemple, si nous avons $I = 3$ médicaments à tester avec 1 médicament de référence (le premier) et 2 nouveaux (les 2 suivants), l'intérêt sera certainement d'estimer l'apport des nouveaux médicaments en comparaison avec le médicament de référence et donc d'estimer 2 différences, $(\mu+\alpha_1)-(\mu+\alpha_2) = \alpha_1-\alpha_2$ et $(\mu+\alpha_1)-(\mu+\alpha_3) = \alpha_1-\alpha_3$. De même, si nous disposons de 2 témoins (les 2 premiers) et de 2 nouveaux médicaments (2 suivants), nous pouvons souhaiter estimer l'apport d'un nouveau médicament en comparaison avec l'effet de référence (*i.e.* la moyenne des 2 témoins). Cela veut dire estimer $(\alpha_1+\alpha_2)/2-\alpha_3$ et $(\alpha_1+\alpha_2)/2-\alpha_4$.
La question est donc : sous quelles conditions une combinaison linéaire des coefficients est-elle estimable de manière unique ? Nous savons qu'il faut que cette combinaison linéaire se trouve dans $\ker(X)^\perp$ mais existe-t-il un critère simple qui assure cela ? C'est l'objet d'un contraste, défini ci-dessous.

Définition 6.2
$\sum_{i=1}^I a_i\alpha_i$ *est un contraste sur les* α_i *si* $\sum_{i=1}^I a_i = 0$.

La définition 6.2 permet de s'assurer que les contrastes sont estimables de manière unique. Les contrastes sont des éléments orthogonaux à $\beta^\dagger$, vecteur de base de $\ker(X)$. En effet nous n'avons pas de contrainte sur μ mais uniquement sur α, c'est-à-dire

$$0 = \sum_{i=1}^I a_i \times 1 = a'\mathbb{1}_I = \langle (0, a')', \beta^\dagger \rangle.$$

Tout vecteur a complété par 0 est donc élément de l'orthogonal de $\ker(X)$ et donc tout contraste est estimable de manière unique.
Nous pouvons vérifier que dans le premier exemple ci-dessus les combinaisons linéaires de coefficients $a = (1, -1, 0)'$ et $b = (1, 0, -1)'$ sont bien des contrastes et donc estimables de manière unique et de même dans le second exemple pour les combinaisons linéaires $a = (1/2, 1/2, -1, 0)'$ et $b = (1/2, 1/2, 0, -1)'$.

Troisième partie

Réduction de dimension

Chapitre 7

Choix de variables

7.1 Introduction

Dans les chapitres précédents, nous avons supposé que le modèle proposé

$$Y = X\beta + \varepsilon$$

pour expliquer Y était juste, et que toutes les variables explicatives $(X_1, \cdots, X_p)$ formant le tableau X étaient importantes dans l'explication de la variable Y.
Dans bon nombre d'études statistiques, nous disposons d'un ensemble de variables explicatives pour potentiellement expliquer la variable réponse (exemple de la concentration de l'ozone). Rien ne nous assure que toutes les variables disponibles interviennent dans l'explication (*i.e.*, dans le modèle de régression). L'utilisateur a donc à sa disposition un ensemble de variables potentiellement explicatives ou variables candidates. Parmi ces variables explicatives, nous incluons les variables originelles ainsi que la transformation de ces variables par des fonctions connues.
Nous supposerons également dans ce chapitre que les données sont de « bonne » qualité, c'est-à-dire qu'il n'y a pas de point aberrant ou levier (voir chapitre 3). En pratique, cette condition est rarement satisfaite.
Nous avons p variables ($p < n$) à notre disposition et nous supposons, comme nous l'avons toujours fait dans ce livre, que la constante (la variable $\mathbb{1}$) fait partie des variables candidates, c'est-à-dire que un des X_j vaut $\mathbb{1}$. Le statisticien peut souhaiter conserver cette variable particulière dans sa modélisation, il aura donc à analyser (2^{p-1}) modèles potentiels. Si par contre la variable $\mathbb{1}$ a le même statut que les autres variables de l'étude, il pourra choisir parmi $(2^p - 1)$ modèles.
Comment alors choisir le meilleur modèle parmi ces modèles ? Pour cela, il faut définir un critère quantifiant la qualité du modèle. Ce critère dépend de l'objectif de la régression. Une fois le critère choisi, il faudra déterminer des procédures permettant de trouver le meilleur modèle. Considérons différents objectifs de la régression et discutons des conséquences sur le choix du modèle.

a. *Estimation des paramètres*
Lorsque les paramètres sont estimés dans des modèles plus petits que le modèle complet (des variables explicatives sont enlevées du modèle complet), les estimateurs obtenus dans ces modèles peuvent être biaisés. En contrepartie, leur variance peut être plus faible que la variance des estimateurs obtenus dans un modèle plus « gros ». Un critère qui combine ces deux effets (biais et variance) est l'erreur quadratique moyenne (EQM) que nous définirons. La trace de l'erreur quadratique moyenne permet de comparer directement des modèles avec un nombre différent de variables explicatives.

Nous pouvons également comparer les modèles *via* l'analyse des valeurs ajustées $\hat{Y}$. Pour chaque modèle, nous obtenons un vecteur de valeurs prédites $\hat{Y}$ dans $\mathbb{R}^n$, et donc, quel que soit le modèle utilisé, nous avons le même objet à analyser.

b. *Sélectionner les variables pertinentes*
L'objectif de la sélection sera alors de déterminer au mieux l'ensemble des variables explicatives X_j tel que leurs coefficients β_j soient non nuls dans le modèle.

c. *Prévision*
Le but de l'étude est de prévoir le mieux possible des nouvelles observations. Pour comparer des modèles sur cette base, nous supposerons que nous recevrons de nouvelles observations notées $(X^\star, Y^\star)$ et nous comparerons l'erreur de prévision obtenue par chaque modèle.

Avant de présenter les différentes procédures et les différents critères de choix, il nous semble important de bien comprendre les conséquences d'un choix erroné de l'ensemble des variables sélectionnées en supposant par ailleurs que cet ensemble existe.

Les notations que nous utilisons sont :

— X est la matrice composée de toutes les variables explicatives $(n \times p)$.
— ξ est un sous-ensemble (d'indices) de $\{1, 2, \ldots, p\}$, son cardinal est noté $|\xi|$. Nous notons $\bar{\xi}$ le complémentaire de ξ dans $\{1, 2, \ldots, p\}$.
— X_ξ est la sous-matrice extraite de X dont les colonnes correspondent aux indices contenus dans ξ.
— Dans le modèle ξ comprenant $|\xi|$ variables, les paramètres associés aux variables sont notés β_ξ et l'estimateur des moindre carrés est désigné par $\hat{\beta}_\xi$.
— Les coordonnées d'indice ξ du vecteur $\hat{\beta}$ sont notées $[\hat{\beta}]_\xi$. En général, $[\hat{\beta}]_\xi \neq \hat{\beta}_\xi$ sauf si $\Im(X_\xi) \perp \Im(X_{\bar{\xi}})$.
— Si nous disposons d'une nouvelle observation $x^{\star\prime} = [x^{\star\prime}_\xi, x^{\star\prime}_{\bar{\xi}}]$, nous avons les prévisions suivantes :

$$\begin{aligned} \hat{y}^p &= x^{\star\prime}\hat{\beta} \\ &= x^{\star\prime}_\xi \hat{\beta}_\xi + x^{\star\prime}_{\bar{\xi}} \hat{\beta}_{\bar{\xi}} \\ &= \hat{y}^p_\xi + x^{\star\prime}_{\bar{\xi}} \hat{\beta}_{\bar{\xi}}. \end{aligned}$$

7.2 Choix incorrect de variables : conséquences

L'objectif de cette section est de bien comprendre les conséquences d'un mauvais choix de variables explicatives. Par « mauvais choix » nous entendons soit en prendre trop peu, soit en prendre le bon nombre mais pas les bonnes, soit en prendre trop. Nous allons analyser un exemple simple et généraliser ensuite les résultats. L'exemple que nous traitons dans cette partie est le suivant : admettons que nous ayons trois variables explicatives potentielles X_1, X_2 et X_3 et que le vrai modèle soit

$$Y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon = X_{12}\beta_{12} + \varepsilon.$$

La variable X_3 n'intervient pas dans le modèle pour prédire Y, mais ce fait n'est pas connu de l'utilisateur de la régression. Avec trois variables, nous pouvons donc analyser sept (2^3-1) modèles différents, trois modèles à une variable, trois modèles à deux variables et un modèle à trois variables. Nous analysons les 7 modèles mais ne précisons les calculs que lorsque $\xi = \{1\}$. Pour ce modèle, nous obtenons comme estimateurs :

$$\begin{aligned}
\hat{\beta}_1 &= (X_1'X_1)^{-1}X_1'Y \\
\hat{Y}_1 &= P_{X_1}Y \\
\hat{\sigma}_1^2 &= \|P_{X_1^\perp}Y\|^2/(n-1).
\end{aligned}$$

7.2.1 Biais des estimateurs

Analysons tout d'abord le biais de ces estimateurs en nous servant du vrai modèle $\mathbb{E}Y = \beta_1 X_1 + \beta_2 X_2 = X_{12}\beta_{12}$. On a

$$\begin{aligned}
\mathbb{E}\hat{\beta}_1 &= (X_1'X_1)^{-1}X_1'\mathbb{E}Y = \beta_1 + (X_1'X_1)^{-1}X_1'X_2\beta_2 \\
\mathbb{E}\hat{Y}_1 &= X_1\beta_1 + P_{X_1}X_2\beta_2.
\end{aligned}$$

Le biais est donc :

$$\begin{aligned}
B(\hat{\beta}_1) &= \mathbb{E}(\hat{\beta}_1) - \beta_1 = (X_1'X_1)^{-1}X_1'X_2\beta_2 \\
B(\hat{Y}_1) &= \mathbb{E}(\hat{Y}_1) - \mathbb{E}(Y) = P_{X_1}X_2\beta_2 - X_2\beta_2 = -P_{X_1^\perp}X_2\beta_2.
\end{aligned}$$

La matrice de projection orthogonale $P_{X_1^\perp}$ est non aléatoire (le choix de X_1 ne se fait pas en fonction de Y), nous pouvons sortir cette matrice de l'espérance. La trace d'un projecteur est égale à la dimension de l'espace sur lequel on projette, nous avons donc :

$$\begin{aligned}
\mathbb{E}\hat{\sigma}_1^2 &= \frac{1}{n-1}\mathbb{E}\operatorname{tr}(Y'P_{X_1^\perp}Y) = \frac{1}{n-1}\operatorname{tr}(P_{X_1^\perp}\mathbb{E}(YY')) \\
&= \frac{1}{n-1}\operatorname{tr}(P_{X_1^\perp}(V(Y) + \mathbb{E}(Y)\mathbb{E}(Y)')) = \sigma^2 + \frac{1}{n-1}\beta_{12}'X_{12}'P_{X_1^\perp}X_{12}\beta_{12} \\
&= \sigma^2 + \frac{1}{n-1}\beta_2^2\|P_{X_1^\perp}X_2\|^2.
\end{aligned}$$

Le biais de la variance estimée vaut ainsi :

$$B(\hat{\sigma}_1^2) \quad = \quad \frac{1}{n-1}\beta_2^2\|P_{X_1^\perp}X_2\|^2.$$

Le biais des estimateurs des 7 modèles possibles est donné dans le tableau 7.1.

modèle	estimations	propriétés
$Y_1 = X_1\beta_1 + \varepsilon$	$\hat{Y}_1 = X_1\hat{\beta}_1$	$B(\hat{Y}_1) = -P_{X_1^\perp}X_2\beta_2$
	$\hat{\sigma}_1^2 = \frac{\|P_{X_1^\perp}Y\|^2}{n-1}$	$B(\hat{\sigma}_1^2) = \frac{1}{n-1}\beta_2^2\|P_{X_1^\perp}X_2\|^2$
$Y = X_2\beta_2 + \varepsilon$	$\hat{Y}_2 = X_2\hat{\beta}_2$	$B(\hat{Y}_2) = -P_{X_2^\perp}X_1\beta_1$
	$\hat{\sigma}_2^2 = \frac{\|P_{X_2^\perp}Y\|^2}{n-1}$	$B(\hat{\sigma}_2^2) = \frac{1}{n-1}\beta_1^2\|P_{X_2^\perp}X_1\|^2$
$Y = X_3\beta_3 + \varepsilon$	$\hat{Y}_3 = X_3\hat{\beta}_3$	$B(\hat{Y}_3) = -P_{X_3^\perp}X_{12}\beta_{12}$
	$\hat{\sigma}_3^2 = \frac{\|P_{X_3^\perp}Y\|^2}{n-1}$	$B(\hat{\sigma}_3^2) = \frac{1}{n-1}\beta_{12}'X_{12}'P_{X_{12}^\perp}X_{12}\beta_{12}$
$Y = X_{12}\beta_{12} + \varepsilon$	$\hat{Y}_{12} = X_{12}\beta_{12}$	$B(\hat{Y}_{12}) = 0$
	$\hat{\sigma}_{12}^2 = \frac{\|P_{X_{12}^\perp}Y\|^2}{n-2}$	$B(\hat{\sigma}_{12}^2) = 0$
$Y = X_{13}\beta_{13} + \varepsilon$	$\hat{Y}_{13} = X_{13}\hat{\beta}_{13}$	$B(\hat{Y}_{13}) = -P_{X_{13}^\perp}X_{12}\beta_{12}$
	$\hat{\sigma}_{13}^2 = \frac{\|P_{X_{13}^\perp}Y\|^2}{n-2}$	$B(\hat{\sigma}_{13}^2) = \frac{1}{n-2}\beta_{12}'X_{12}'P_{X_{13}^\perp}X_{12}\beta_{12}$
$Y = X_{23}\beta_{23} + \varepsilon$	$\hat{Y}_{23} = X_{23}\hat{\beta}_{23}$	$B(\hat{Y}_{23}) = -P_{X_{23}^\perp}X_{12}\beta_{12}$
	$\hat{\sigma}_{23}^2 = \frac{\|P_{X_{23}^\perp}Y\|^2}{n-2}$	$B(\hat{\sigma}_{23}^2) = \frac{1}{n-2}\beta_{12}'X_{12}'P_{X_{23}^\perp}X_{12}\beta_{12}$
$Y = X_{123}\beta_{123} + \varepsilon$	$\hat{Y}_{123} = X_{123}\hat{\beta}_{123}$	$B(\hat{Y}_{123}) = 0$
	$\hat{\sigma}_{123}^2 = \frac{\|P_{X_{123}^\perp}Y\|^2}{n-3}$	$B(\hat{\sigma}_{123}^2) = 0$

Tableau 7.1 – Biais des différents estimateurs.

Nous constatons alors que dans les modèles « trop petits » (ici à 1 variable), c'est-à-dire admettant moins de variables que le modèle « correct » inconnu du statisticien, les estimateurs obtenus sont biaisés. A l'inverse, lorsque les modèles sont « trop grands » (ici à 3 variables), les estimateurs ne sont pas biaisés. Il semblerait donc qu'il vaille mieux travailler avec des modèles « trop grands ». Nous pouvons énoncer un résultat général (voir exercice 7.2).

Proposition 7.1
$\hat{\beta}_\xi$ et $\hat{Y}_\xi$ sont en général biaisés.

L'estimation du biais est difficile car $x'\beta$ est inconnue. Remarquons que $\hat{\sigma}_\xi^2$ est en général biaisé positivement, c'est-à-dire que, en moyenne, l'espérance de $\hat{\sigma}_\xi^2$ vaut σ^2 plus une quantité positive.

7.2.2 Variance des estimateurs

Les dimensions des estimateurs varient avec la taille du modèle. Cependant, en nous servant de la formule d'inverse par bloc donnée en annexe, nous pouvons montrer que les estimateurs des composantes communes ont des variances plus faibles dans le modèle le plus petit :

$$\mathrm{V}(\hat{\beta}_1) \leq \mathrm{V}([\hat{\beta}_{12}]_1) \leq \mathrm{V}([\hat{\beta}_{123}]_1).$$

où

$$\begin{array}{rrcl}
Y = X_1\beta_1 + \varepsilon & V(\hat{\beta}_1) & = & (X_1'X_1)^{-1}\sigma^2 \\
Y = X_{12}\beta_{12} + \varepsilon & V(\hat{\beta}_{12}) & = & \begin{pmatrix} X_1'X_1 & X_1'X_2 \\ & X_2'X_2 \end{pmatrix}\sigma^2 \\
Y = X_{123}\beta_{123} + \varepsilon & V(\hat{\beta}_{123}) & = & \begin{pmatrix} X_1'X_1 & X_1'X_2 & X_1'X_3 \\ & X_2'X_2 & X_2'X_3 \\ & & X_3'X_3 \end{pmatrix}\sigma^2.
\end{array}$$

Si nous travaillons avec les valeurs ajustées, nous avons le même phénomène :

$$\begin{array}{rrcl}
Y = X_1\beta_1 + \varepsilon & V(\hat{Y}_1) & = & P_{X_1}\sigma^2 \\
Y = X_{12}\beta_{12} + \varepsilon & V(\hat{Y}_{12}) & = & P_{X_{12}}\sigma^2 = P_{X_1}\sigma^2 + P_{X_2\cap X_1^\perp}\sigma^2 \\
Y = X_{123}\beta_{123} + \varepsilon & V(\hat{Y}_{123}) & = & P_{X_{123}}\sigma^2 = P_{X_1}\sigma^2 + P_{X_{23}\cap X_1^\perp}\sigma^2.
\end{array}$$

Nous pouvons énoncer un résultat général (voir exercice 7.3) :

Proposition 7.2
1. $\mathrm{V}([\hat{\beta}]_\xi) - \mathrm{V}(\hat{\beta}_\xi)$ *est une matrice semi-définie positive, ce qui veut dire que les composantes communes aux deux modèles sont mieux estimées (moins variables) dans le modèle le plus petit.*

2. La variance des données ajustées dans le modèle le plus petit est plus faible que celle des données ajustées dans le modèle plus grand $\mathrm{V}(\hat{Y}) \geq \mathrm{V}(\hat{Y}_\xi)$.

Si le critère de choix de modèle est la variance, l'utilisateur choisira des modèles admettant zéro paramètre à estimer ! En général, il est souhaitable d'obtenir un modèle précis en moyenne (faible biais) et ayant une variance faible. Nous venons de voir qu'un moyen simple d'atteindre le premier objectif consiste à conserver toutes les variables dont nous disposons alors que le second sera atteint en éliminant toutes les variables. L'erreur quadratique moyenne (EQM) va concilier ces deux objectifs.Cette définition a été donnée au chapitre 2 mais nous la rappelons ici.

7.2.3 Erreur quadratique moyenne

L'erreur quadratique moyenne (EQM) d'un estimateur $\hat{\theta}$ de θ de dimension p est

$$\begin{array}{rcl}
\mathrm{EQM}(\hat{\theta}) & = & \mathbb{E}((\theta - \hat{\theta})(\theta - \hat{\theta})') \\
& = & \mathbb{E}(\theta - \hat{\theta})\mathbb{E}(\theta - \hat{\theta})' + V(\hat{\theta}),
\end{array}$$

c'est-à-dire le biais « au carré » plus la variance. Un estimateur biaisé peut être meilleur qu'un estimateur non biaisé si sa variance est plus petite.
Revenons au problème de la régression où nous avons plusieurs ensembles de variables ξ. Nous allons utiliser l'EQM comme mesure de comparaison. Nous pouvons comparer soit des estimateurs $\hat{\beta}_\xi \in \mathbb{R}^{|\xi|}$, soit des valeurs ajustées $x'_\xi \hat{\beta}_\xi \in \mathbb{R}$, où x'_ξ correspond à une ligne de la matrice X_ξ, soit des valeurs prévues $x^{\star\prime}_\xi \hat{\beta}_\xi \in \mathbb{R}$, où $x^\star_\xi \in \mathbb{R}^{|\xi|}$ est une nouvelle observation. Il est classique de traiter le choix de variables *via* l'analyse de la valeur ajustée $\hat{y}$ ou de la valeur prévue $\hat{y}^p$ et non pas *via* les estimateurs $\hat{\beta}_\xi$ dont les dimensions varient avec $|\xi|$. Les définitions que nous allons introduire de l'EQM et de l'EQM de prévision, notée EQMP, seront adaptées à notre problème.

Définition 7.1 (EQM)
Considérons le modèle de régression $Y = X\beta + \varepsilon$ où β, le paramètre inconnu du modèle, peut avoir des coordonnées nulles. Soit $x \in \mathbb{R}^p$ le vecteur colonne d'une observation, nous avons $x_\xi \in \mathbb{R}^{|\xi|}$ et $\hat{\beta}_\xi$ l'estimateur des MC obtenus avec ces $|\xi|$ variables. L'erreur quadratique moyenne est définie par

$$\mathrm{EQM}(\hat{y}_\xi) = \mathbb{E}((x'_\xi \hat{\beta}_\xi - x'\beta)^2) = \mathrm{V}(x'_\xi \hat{\beta}_\xi) + B^2(x'_\xi \hat{\beta}_\xi),$$

où $B(x'_\xi \hat{\beta}_\xi) = \mathbb{E}(x'_\xi \hat{\beta}_\xi) - x'\beta$ est le biais de $x'_\xi \hat{\beta}_\xi$.
Si nous possédons n observations x_ξ regroupées dans une matrice X_ξ et $\hat{\beta}_\xi$ l'estimateur des MC obtenu avec ces $|\xi|$ variables, nous définissons la trace de la matrice de l'EQM par

$$\mathrm{tr}(\mathrm{EQM}(\hat{Y}_\xi)) = \mathrm{tr}(\mathrm{V}(X_\xi \hat{\beta}_\xi)) + B(X_\xi \hat{\beta}_\xi)' B(X_\xi \hat{\beta}_\xi).$$

Nous pouvons développer le calcul de la décomposition de l'EQM pour les valeurs ajustées avec le modèle ξ

$$\begin{aligned} \mathrm{tr}(\mathrm{EQM}(\hat{Y}_\xi)) &= \mathrm{tr}(\mathrm{V}(X_\xi \hat{\beta}_\xi)) + B(X_\xi \hat{\beta}_\xi)' B(X_\xi \hat{\beta}_\xi) \\ &= \mathrm{tr}(\mathrm{V}(P_{X_\xi} Y)) + (\mathbb{E}(X_\xi \hat{\beta}_\xi) - X\beta)'(\mathbb{E}(X_\xi \hat{\beta}_\xi) - X\beta) \\ &= |\xi|\sigma^2 + \|(I - P_{X_\xi})X\beta\|^2. \end{aligned} \tag{7.1}$$

Afin de pouvoir sortir P_{X_ξ} de la variance, il faut que P_{X_ξ} soit fixe et donc que *le choix du modèle X_ξ ne dépende pas des données sur lesquelles on évalue le projecteur*. Si le choix des variables a été effectué sur le même jeu de données que celui qui sert à estimer les paramètres, nous devrions considérer un terme de biais supplémentaire appelé biais de sélection. Nous reviendrons sur ce concept à la fin du chapitre.
Revenons à l'exemple et calculons l'EQM des 7 modèles

$$Y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon = X_{12}\beta_{12} + \varepsilon.$$

Considérons le modèle avec une variable X_1, nous avons pour le terme tr(EQM),

utilisant $\mathcal{H}_2$ et des propriétés des projecteurs (symétrie, idempotence et trace) :

$$\begin{aligned}
\mathrm{tr}(\mathrm{EQM}(X_1\hat{\beta}_1)) &= \mathrm{tr}(\mathrm{V}(X_1\hat{\beta}_1)) + B(X_1\hat{\beta}_1)'B(X_1\hat{\beta}_1) \\
&= \mathrm{tr}(\mathrm{V}(P_{X_1}Y)) + \|\mathbb{E}(X_1\hat{\beta}_1) - X_{12}\beta_{12}\|^2 \\
&= \sigma^2\,\mathrm{tr}(P_{X_1}) + \|\mathbb{E}(P_{X_1}(X_{12}\beta_{12}+\varepsilon)) - X_{12}\beta_{12}\|^2 \\
&= \sigma^2 + \|P_{X_1^\perp}X_{12}\beta_{12}\|^2.
\end{aligned}$$

Nous avons donc :

$$\begin{aligned}
\mathrm{tr}(\mathrm{EQM}(X_1\hat{\beta}_1)) &= \sigma^2 + \|P_{X_1^\perp}X_{12}\beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQM}(X_2\hat{\beta}_2)) &= \sigma^2 + \|P_{X_2^\perp}X_{12}\beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQM}(X_3\hat{\beta}_3)) &= \sigma^2 + \|P_{X_3^\perp}X_{12}\beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQM}(X_{12}\hat{\beta}_{12})) &= 2\sigma^2 \\
\mathrm{tr}(\mathrm{EQM}(X_{13}\hat{\beta}_{13})) &= 2\sigma^2 + \|P_{X_{13}^\perp}X_{12}\beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQM}(X_{23}\hat{\beta}_{23})) &= 2\sigma^2 + \|P_{X_{23}^\perp}X_{12}\beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQM}(X_{123}\hat{\beta}_{123})) &= 3\sigma^2.
\end{aligned}$$

Dans le cas où nous connaissons le bon modèle, choisir le modèle ayant la plus petite tr(EQM) parmi les sept modèles initiaux revient à analyser la tr(EQM) des quatre modèles suivants :

$$\mathrm{tr}(\mathrm{EQM}(X_1\hat{\beta}_1)), \quad \mathrm{tr}(\mathrm{EQM}(X_2\hat{\beta}_2)), \quad \mathrm{tr}(\mathrm{EQM}(X_3\hat{\beta}_3)) \quad \text{et} \quad \mathrm{tr}(\mathrm{EQM}(X_{12}\hat{\beta}_{12})).$$

Supposons maintenant que nous connaissons les autres quantités inconnues et que la plus petite norme soit celle de $\|P_{X_1^\perp}X_{12}\beta_{12}\|^2$. Il nous faut donc choisir entre

$$\mathrm{tr}(\mathrm{EQM}(X_1\hat{\beta}_1)) = \sigma^2 + \|P_{X_1^\perp}X_{12}\beta_{12}\|^2 \quad \text{et} \quad \mathrm{tr}(\mathrm{EQM}(X_{12}\hat{\beta}_{12})) = 2\sigma^2.$$

Afin de choisir le modèle ayant la plus petite tr(EQM), il faut comparer σ^2 à $\|P_{X_1^\perp}X_{12}\beta_{12}\|^2$. Cela sera donc le modèle X_1 ou le modèle X_{12}, tout dépendra de la valeur de σ^2 et de $\|P_{X_1^\perp}X_{12}\beta_{12}\|^2$. Dans l'exemple de la figure 7.1, nous sélectionnons le modèle 2 (le vrai modèle) car dans ce cas $\|P_{X_1^\perp}X_{12}\beta_{12}\|^2 > \sigma^2$. Si au contraire $\|P_{X_1^\perp}X_{12}\beta_{12}\|^2 < \sigma^2$, nous choisissons le modèle 1, c'est-à-dire un modèle un peu faux (le terme de biais non nul) mais plus précis (la variance est plus faible) que le vrai modèle.

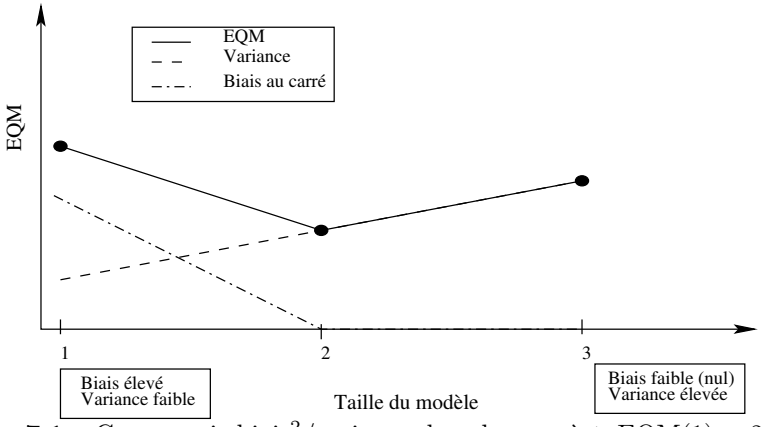


Fig. 7.1 – Compromis biais^2/variance dans la cas où $\operatorname{tr}\mathrm{EQM}(1) > 2\sigma^2$.

Il est en général difficile d'estimer le biais car la valeur du paramètre est inconnue, il est par contre plus facile d'estimer la variance. Nous verrons dans la suite de ce chapitre des procédures pour estimer l'EQM, mais dans un premier temps il semble plus facile de considérer l'EQM de prévision ou sa trace.

7.2.4 Erreur quadratique moyenne de prévision

L'EQM ou sa trace est un critère classique en statistique, mais il ne fait pas intervenir de nouvelles observations $Y^\star$. Si l'on souhaite donc évaluer l'EQM de prévision de ces nouvelles observations $Y^\star$, nous avons la définition suivante :

Définition 7.2 (EQMP)
Considérons $x^\star \in \mathbb{R}^p$, une nouvelle observation, et $x^\star_\xi$ ses composantes correspondant à ξ. L'erreur quadratique moyenne de prévision est définie par

$$\mathrm{EQMP}(\hat{y}^p_\xi) = \mathbb{E}((x^{\star\prime}_\xi \hat{\beta}_\xi - y^\star)^2) = \mathrm{EQM}(x^{\star\prime}_\xi \hat{\beta}_\xi) + \sigma^2 - 2\mathbb{E}([x^{\star\prime}_\xi \hat{\beta}_\xi - x^{\star\prime}\beta]\varepsilon^\star).$$

Si $\varepsilon^\star$ n'est pas corrélé avec les ε, nous avons alors

$$\mathrm{EQMP}(\hat{y}^p_\xi) = \mathrm{EQM}(x^\star_\xi \hat{\beta}_\xi) + \sigma^2.$$

*Si nous possédons $n^\star$ nouvelles observations $x^\star$ regroupées dans une matrice $X^\star$, nous utilisons la trace de l'*EQMP

$$\operatorname{tr}(\mathrm{EQMP}(\hat{Y}^p_\xi)) = \operatorname{tr}(\mathrm{EQM}(X^\star_\xi \hat{\beta})) + n^\star\sigma^2 - 2\mathbb{E}((X^\star_\xi \hat{\beta}_\xi - X^\star\beta)'\varepsilon^\star).$$

Si $\varepsilon^\star$ n'est pas corrélé avec les ε, nous avons alors

$$\operatorname{tr}(\mathrm{EQMP}(\hat{Y}^p_\xi)) = \operatorname{tr}(\mathrm{EQM}(X^\star_\xi \hat{\beta}_\xi)) + n^\star\sigma^2.$$

La dernière équation ci-dessus nous indique la chose suivante : si les données sur lesquelles se fait la prévision sont indépendantes des données sur lesquelles sont

calculées les estimations (deux jeux de données différents), alors l'EQM et l'EQMP sont identiques à ($n^\star$ fois) la variance de l'erreur près.
Dans le cas où les données X sont utilisées pour l'estimation et la prévision (dans ce cas-là il est d'usage de parler plutôt d'ajustement). La formule de l'EQMP devient :

$$\begin{aligned}
\mathrm{tr}(\mathrm{EQMP}(\hat{Y})) &= \mathbb{E}\|\hat{Y} - Y\|^2 \\
&= \mathrm{tr}(\mathrm{EQM}(X\hat{\beta})) + n\sigma^2 - 2\mathbb{E}(\langle X\hat{\beta} - X\beta, \varepsilon\rangle) \\
&= \mathrm{tr}(\mathrm{EQM}(X\hat{\beta})) + n\sigma^2 - 2\mathbb{E}(\langle X\hat{\beta}, \varepsilon\rangle) \\
&= \mathrm{tr}(\mathrm{EQM}(X\hat{\beta})) + n\sigma^2 - 2\mathbb{E}(\varepsilon' P_X \varepsilon) \\
&= \mathrm{tr}(\mathrm{EQM}(X\hat{\beta})) + n\sigma^2 - 2p\sigma^2.
\end{aligned}$$

Reprenons maintenant l'exemple précédent

$$Y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon = X_{12}\beta_{12} + \varepsilon$$

et supposons que nous ayons $n^\star$ nouvelles observations concaténées dans la matrice $X^\star$. Nous avons alors

$$\begin{aligned}
\mathrm{tr}(\mathrm{EQMP}(X_1^\star\hat{\beta}_1)) &= (n^\star + 1)\sigma^2 + \|P_{X_1^\perp} X_{12}^\star \beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_2^\star\hat{\beta}_2)) &= (n^\star + 1)\sigma^2 + \|P_{X_2^\perp} X_{12}^\star \beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_3^\star\hat{\beta}_3)) &= (n^\star + 1)\sigma^2 + \|P_{X_3^\perp} X_{12}^\star \beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_{12}^\star\hat{\beta}_{12})) &= (n^\star + 2)\sigma^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_{13}^\star\hat{\beta}_{13})) &= (n^\star + 2)\sigma^2 + \|P_{X_{13}^\perp} X_{12}^\star \beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_{23}^\star\hat{\beta}_{23})) &= (n^\star + 2)\sigma^2 + \|P_{X_{23}^\perp} X_{12}^\star \beta_{12}\|^2 \\
\mathrm{tr}(\mathrm{EQMP}(X_{123}^\star\hat{\beta}_{123})) &= (n^\star + 3)\sigma^2.
\end{aligned}$$

Si nous calculons la tr(EQMP) théorique des trois modèles, nous obtenons

$$\begin{aligned}
\mathrm{tr}(\mathrm{EQMP}(\hat{Y}(X_1)) &= \|P_{X_1^\perp} X\beta\|^2 + \sigma^2(n-1) \\
\mathrm{tr}(\mathrm{EQMP}(\hat{Y}(X_{12})) &= \sigma^2(n-2) \\
\mathrm{tr}(\mathrm{EQMP}(\hat{Y}(X_{123}))) &= \sigma^2(n-3).
\end{aligned}$$

La tr(EQMP) préconise d'utiliser le modèle ayant le plus de variables explicatives. En fait ce critère n'a pas de sens lorsqu'il est utilisé sur les données qui ont servi à estimer les paramètres.
Nous venons de voir les effets d'un mauvais choix de variables et il est donc important voire même primordial d'effectuer correctement ce choix en pratique. Pour comparer ces modèles, nous pouvons utiliser la validation croisée qui est présentée en détails algorithme 3 p. 238. Le principe est toujours le même, à savoir qu'on divise le jeu de données initial en K parties distinctes approximativement de même taille. Pour une partie donnée, par exemple la i^{e}, on met de côté cette i^{e} partie

des données pour effectuer la prédiction après avoir estimé les modèles sur toutes les autres observations appelées souvent données d'apprentissage. Et on répète ce travail sur les K parties. Ainsi à la fin de la procédure, tous les individus ont été prévus une fois et comme nous connaissons la valeur de la variable à expliquer, il est donc possible d'évaluer la qualité de la prévision. Le critère utilisé en général est l'erreur quadratique moyenne de prévision (EQMP) qui est la moyenne des erreurs de prévision au carrés

$$EQM = \frac{1}{n}\sum_{i=1}^{n}(Y_i^p - Y)^2,$$

Y_i^p désigne la prévision de l'observation i sans avoir utilisé l'observation i dans le modèle. Le modèle sélectionné est bien sûr le modèle $\tilde{\xi}$ qui minimise cette moyenne. La procédure est une procédure de validation croisée de taille K (*K-fold cross-validation*). Il est possible de faire non pas K blocs mais n blocs, cela correspond à laisser de côté un individu à chaque fois et cette procédure est appelée (*leave-one-out*).
En général, l'ordre de grandeur de K est 10, si le nombre d'observations par bloc est suffisant. Mais pour optimiser le temps de calcul, ce choix peut être un multiple du nombre de coeurs de votre ordinateur car cette procédure se parallélise facilement. Le problème réside dans le fait qu'il faille calculer le critère de choix, sur tous les ensembles de variables ξ. Cette procédure n'étant pas implémentée dans les logiciels, nous allons présenter les critères classiques de choix de variables et nous montrerons dans le chapitre 10 comment combiner les méthodes de sélection avec cette méthode de comparaison.

7.3 Critères classiques de choix de modèles

Nous allons nous intéresser aux méthodes classiques de sélection de modèle. Les principaux critères de choix sont le R^2, le R_a^2, le C_p, l'AIC, le BIC et leurs extensions. D'un autre côté, le test F entre modèles emboîtés permet de comparer selon une approche de type test classique les modèles entre eux. Quand ceux-ci ne sont pas emboîtés l'un dans l'autre, une approche basée sur des intervalles de confiance peut être utilisée. Cette approche, moins répandue, n'est pas en général implémentée dans les logiciels. Néanmoins le lecteur intéressé pourra consulter la description de Miller (2002).

Nous allons présenter différents critères de choix de modèles et les appliquer aux données de l'ozone. Il y a donc $n = 50$ observations, la constante sera toujours dans le modèle et nous avons 9 variables explicatives potentielles. Sur ce jeu de données, nous pouvons analyser 512 (2^9) modèles (la constante est dans tous les modèles).

7.3.1 Tests entre modèles emboîtés

Si les modèles concurrents sont emboîtés les uns dans les autres, il est alors possible d'utiliser une procédure de test (5.2 p. 100). Notons le modèle ξ à $|\xi|$ variables et le modèle ξ_{+1} correspondant au modèle ξ auquel on a rajouté une variable supplémentaire. Afin de choisir entre ces deux modèles emboîtés, nous avons la statistique de test suivante (voir p. 100) :

$$F = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi_{+1})}{\hat{\sigma}^2}.$$

Afin que F suive une loi de Fisher, l'estimation de $\hat{\sigma}^2$ doit suivre une loi du χ^2 indépendante du numérateur. Classiquement σ^2 est estimé de deux manières différentes :

1. Estimation de σ^2 par $\mathrm{SCR}(\xi_{+1})/(n - |\xi| - 1)$.
Si on cherche à choisir seulement entre ξ et ξ_{+1}, alors l'estimateur utilisé pour σ^2 est celui provenant du modèle le plus « grand », soit le modèle (ξ_{+1}). Cette solution est en général utilisée par les logiciels de statistiques.

2. Estimation de σ^2 par $\mathrm{SCR}(p)/(n - p)$.
Si on considère tous les modèles possibles construits à partir de p variables explicatives, alors l'estimateur pour la variance des résidus σ^2 utilisé provient de l'estimateur trouvé pour le modèle complet.

Nous avons donc le théorème suivant.

Théorème 7.1 (Tests entre modèles emboîtés)
Soient deux modèles, le modèle ξ et le modèle ξ_{+1}. Le test permettant de tester l'hypothèse $\mathrm{H}_0 : \quad \mathbb{E}Y \in \mathcal{M}_{X_\xi}$ *contre l'hypothèse* $\mathrm{H}_1 : \quad \mathbb{E}Y \in \mathcal{M}_{X_{\xi_{+1}}}$, *s'effectue selon le protocole suivant (en fonction de l'estimation de σ^2).*

1. *La variance σ^2 est estimée par* $\mathrm{SCR}(\xi_{+1})/(n - |\xi| - 1)$. *Si*

$$F_1 = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi_{+1})}{\mathrm{SCR}(\xi_{+1})} \times (n - |\xi| - 1) > f_{1,n-|\xi|-1}(1 - \alpha)$$

alors le modèle ξ est repoussé, au niveau α du test, au profit du modèle (ξ_{+1}), une variable est rajoutée au modèle.

2. *La variance σ^2 est estimée par* $\mathrm{SCR}(p)/(n - p)$. *Si*

$$F_2 = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi_{+1})}{\mathrm{SCR}(p)} \times (n - p) > f_{1,n-p}(1 - \alpha).$$

alors le modèle ξ est repoussé, au niveau α du test, au profit du modèle (ξ_{+1}), une variable est rajoutée au modèle.

Si ξ_{+1} est le vrai modèle, alors les deux estimateurs de σ^2, $\mathrm{SCR}(\xi_{+1})$ et $\mathrm{SCR}(p)$ sont non-biaisés, mais la variance de $\mathrm{SCR}(\xi_{+1})$ est plus petite que la variance de $\mathrm{SCR}(p)$. Donc dans ce cas, on préfère le test F_1 car il est plus puissant. Si aucun des deux modèles n'est le vrai modèle, il est difficile de comparer ces deux tests car σ^2 n'est pas estimée de la même manière.

7.3.2 Le R^2

Le R^2 est défini *via* la SCR, en effet

$$\mathrm{R}^2(\xi) = \frac{\|\hat{Y}_{|\xi|} - \bar{y}\mathbb{1}\|^2}{\|Y - \bar{y}\mathbb{1}\|^2} = 1 - \frac{\mathrm{SCR}(\xi)}{\mathrm{SCT}}.$$

Il s'agit d'un critère relié directement à la $\mathrm{SCR}(\xi)$). Le R^2 augmente toujours avec le nombre de variables $|\xi|$. Comparons la variation du $\mathrm{R}^2(\xi)$ obtenu avec les ξ variables et le R^2 obtenu avec les mêmes ξ variables plus une autre variable, soit $\mathrm{R}^2(\xi_{+1})$.

$$\begin{aligned}
\mathrm{R}^2(\xi_{+1}) - \mathrm{R}^2(\xi) &= \frac{\|P_{X_\xi^\perp} Y\|^2 - \|P_{X_{\xi_{+1}}^\perp} Y\|^2}{\|Y - \bar{y}\mathbb{1}\|^2} \\
&= \frac{\|P_{X_\xi^\perp} P_{X_{\xi_{+1}}^\perp} Y + P_{X_\xi^\perp} P_{X_{\xi_{+1}}} Y\|^2 - \|P_{X_{\xi_{+1}}^\perp} Y\|^2}{\|Y - \bar{y}\mathbb{1}\|^2} \\
&= \frac{\|P_{X_\xi^\perp} P_{X_{\xi_{+1}}} Y\|^2}{\|Y - \bar{y}\mathbb{1}\|^2} \geq 0.
\end{aligned}$$

Nous avons le graphique général suivant :

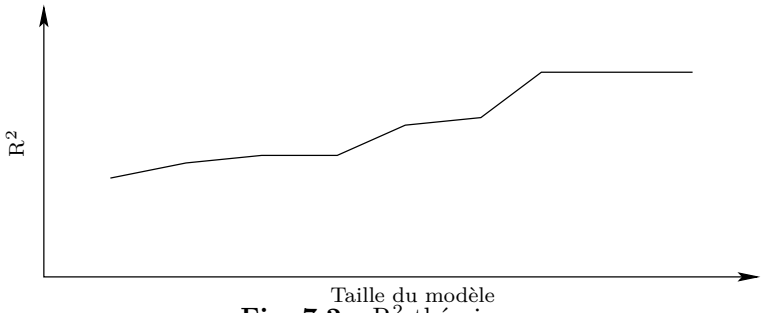


Fig. 7.2 – R^2 théorique.

Bien entendu le même résultat est obtenu avec la définition du R^2 quand les deux modèles ne contiennent pas la constante (2.4, p. 42).

En général, il ne faut donc pas utiliser le R^2 *comme critère de choix de modèle car ce critère va toujours augmenter avec le nombre de variables. Il peut cependant servir à comparer des modèles ayant le même nombre de variables explicatives.*

Voyons cela sur l'exemple de l'ozone :

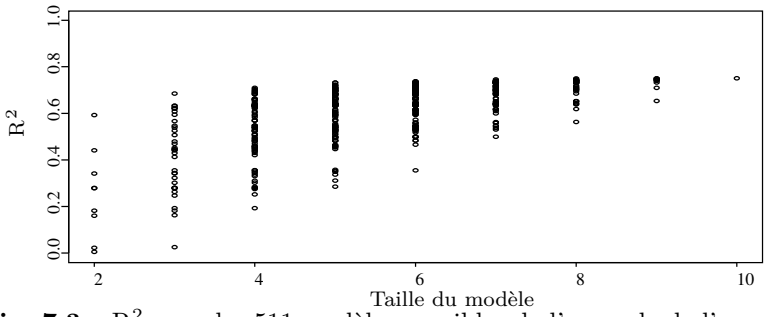


Fig. 7.3 – R^2 pour les 511 modèles possibles de l'exemple de l'ozone.

Nous savons que cette quantité croît avec le nombre de variables incluses dans le modèle et ce résultat se retrouve sur le graphique (fig. 7.3). Le R^2 ne permet pas de choisir entre différents modèles. De manière classique on parle alors d'ajustement de qualité croissante des données : le R^2 augmente, la SCR diminue, donc l'erreur estimée est de plus en plus petite et donc les ajustements $\hat{Y}$ sont de plus en plus proches de Y. On ne parle pas de prévision puisqu'on a utilisé les Y pour estimer $\hat{Y}$. Par contre, à taille fixée, le R^2 permet de comparer les modèles entre eux et de sélectionner celui qui donne le meilleur ajustement.

En considérant le graphique 7.3, le meilleur modèle au sens du R^2 est donc celui avec 10 variables. Cependant, la valeur du R^2 obtenue pour le meilleur modèle à 5 variables est relativement proche de la valeur du R^2 obtenue avec le modèle complet. L'utilisateur pourra peut-être considérer ce modèle.

7.3.3 Le R^2 ajusté

Le R^2 ajusté est défini par

$$\begin{aligned} R_a^2(\xi) &= 1 - \frac{n-1}{n-|\xi|}\left(1 - R^2(\xi)\right) \\ &= 1 - \frac{\mathrm{SCR}(\xi)/(n-|\xi|)}{\mathrm{SCT}\,/(n-1)} \\ &= 1 - \frac{n-1}{\mathrm{SCT}}\frac{\mathrm{SCR}(\xi)}{n-|\xi|}. \end{aligned}$$

Le R_a^2 est donc fonction des carrés moyens définis comme la somme des carrés divisée par le nombre de degrés de liberté. Le but est de maximiser le R_a^2, ce qui revient à minimiser $\mathrm{SCR}(\xi)$ divisée par son degré de liberté. La SCR et $n - |\xi|$ diminuent lorsque $|\xi|$ augmente. Le carré moyen résiduel $\mathrm{CMR}(\xi)$ peut augmenter lorsque la réduction de la SCR, obtenue en ajoutant une variable dans le modèle, ne suffit pas à compenser la perte d'un ddl du dénominateur. Nous obtenons alors en général le graphique suivant pour la SCR $/ddl$ et le R_a^2 ajusté :

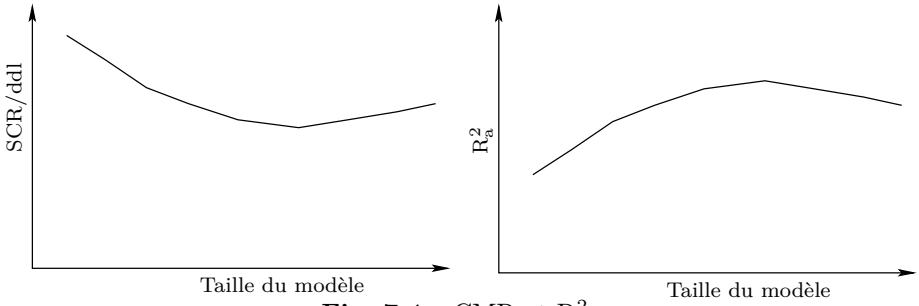


Fig. 7.4 – CMR et R_a^2.

Voyons maintenant le critère du R_a^2 sur l'exemple de l'ozone :

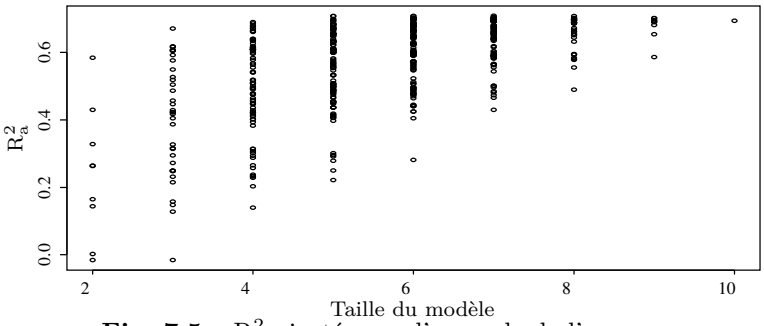


Fig. 7.5 – R^2 ajusté pour l'exemple de l'ozone.

Sur le graphique précédent, l'utilisation du R_a^2 nous conduirait à choisir un modèle à 5 ou 6 variables.

7.3.4 Le C_p de Mallows

La définition du C_p de Mallows (1973) est la suivante :

Définition 7.3
Le $C_p(\xi)$ *d'un modèle à* ξ *variables explicatives est défini par*

$$C_p(\xi) = \frac{SCR(\xi)}{\hat{\sigma}^2} - n + 2|\xi|, \tag{7.2}$$

où SCR *est la valeur de la* $SCR(\xi)$ *dans le sous-modèle caractérisé par* ξ *alors que* $\hat{\sigma}^2$ *est un estimateur sans biais de* σ^2. *En général* $\hat{\sigma}^2$ *a été estimé dans le modèle complet à* p *variables.*

Remarque
Si P_{X_ξ} est non aléatoire (le choix de ξ ne dépend pas de Y), nous avons montré (équation (7.1)) que

$$\mathrm{tr}(\mathrm{EQM}(\hat{Y}_\xi)) \quad = \quad |\xi|\sigma^2 + \|(I - P_{X_\xi})X\beta\|^2.$$

Calculons l'espérance de la somme des carrés résiduels :

$$\begin{aligned}\mathbb{E}(\mathrm{SCR}(\xi)) &= \mathbb{E}(\|Y-\hat{Y}_\xi\|^2)\\ &= \mathbb{E}(\|(I-P_{X_\xi})X\beta+(I-P_{X_\xi})\varepsilon\|^2)\\ &= \|(I-P_{X_\xi})X\beta\|^2+(n-|\xi|)\sigma^2.\end{aligned}$$

En remplaçant, nous obtenons

$$\mathrm{tr}(\mathrm{EQM}(\hat{Y}_\xi)) = \mathbb{E}(\mathrm{SCR}(\xi))-(n-2|\xi|)\sigma^2$$

A modèle fixé, $\hat{\sigma}^2\,\mathrm{C_p}$ est un estimateur sans biais de la trace de l'EQM . Intuitivement le modèle avec le $\hat{\sigma}^2\,\mathrm{C_p}$ le plus faible sera (en moyenne du moins) le modèle avec la tr(EQM) la plus faible et donc la tr(EQMP) la plus faible. Cependant, outre les hypothèses classiques (indépendance du bruit, homoscédasticité et X fixé) afin d'avoir l'égalité $\mathbb{E}(P_{X_\xi}Y) = P_{X_\xi}\mathbb{E}(Y)$ utilisée dans le calcul, il faudrait que *le choix du modèle X_ξ ne dépende pas des données sur lesquelles on évalue le* $\mathrm{C_p}$. Autrement dit, pour que le $\mathrm{C_p}$ ou plus exactement $\hat{\sigma}^2\,\mathrm{C_p}$ soit un bon estimateur de l'EQM, il faut que l'estimation des paramètres et le choix des modèles ne dépendent pas de données sur lesquelles on calcule le $\hat{\sigma}^2\,\mathrm{C_p}$. Cela est rarement fait en pratique et donc l'estimateur du $\mathrm{C_p}$ est biaisé. Ce biais est appelé **biais de sélection** et nous étudierons en détail ce phénomène en fin de chapitre.

Dessiner le $\mathrm{C_p}(\xi)$

En général, nous dessinons en abscisse la valeur de $|\xi|$ et en ordonnée la valeur correspondante de $\mathrm{C_p}(\xi)$ pour tous les modèles. Ce dessin est en général peu lisible et on préfère retenir le meilleur modèle à ξ variables et dessiner les p valeurs de $\mathrm{C_p}(\xi)$ en fonction de $|\xi|$ (fig. 7.6).

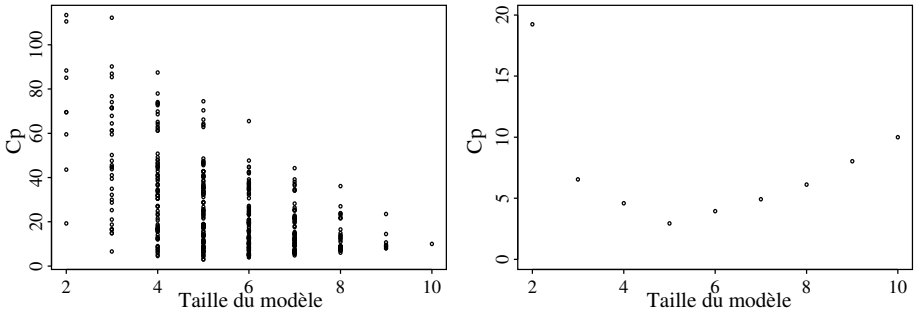


Fig. 7.6 – Choix du $\mathrm{C_p}$ pour l'exemple de l'ozone, 511 modèles, ou meilleur modèle pour chaque taille possible.

Choisir le modèle grâce au $\mathrm{C_p}(\xi)$ et interpréter

Classiquement, il est recommandé de choisir le modèle admettant

$$\mathrm{C_p}(\xi) \leq |\xi|.$$

Le choix du modèle *via* le $C_p(\xi)$ sera le modèle dont la valeur du $C_p(\xi)$ sera proche de la première bissectrice ($y = |\xi|$).

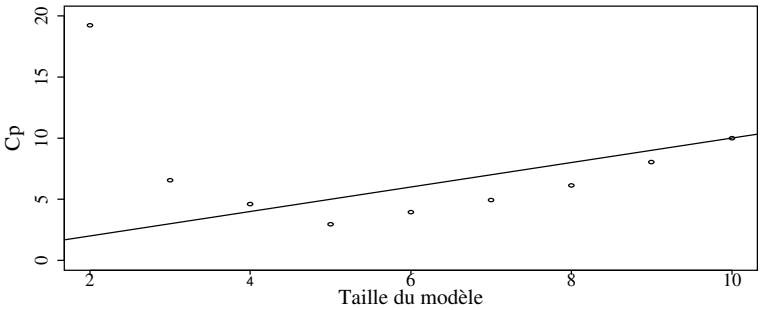


Fig. 7.7 – Choix du C_p pour l'exemple de l'ozone.

Au vu de ce graphique, les modèles admettant plus de 4 variables sont susceptibles d'être sélectionnés.

Interprétation
Plus le modèle est explicatif, plus la quantité $\mathrm{SCR}(\xi)$ est faible. Cette quantité diminue si l'on ajoute des variables à un modèle donné puisque l'on projette sur des sous-espaces de taille croissante. Le critère C_p permet donc un équilibre entre un faible nombre de variables ($|\xi|$ faible) et une $\mathrm{SCR}(\xi)$ faible. Il est possible de généraliser le C_p en remplaçant le coefficient 2 qui assure la « balance » par une fonction des données notée $f(n)$ qui soit différente de 2.
Si le modèle est correct (si les variables intervenant dans le modèle ont été sélectionnées sans utiliser les données), alors $\mathrm{SCR}(\xi)$ est un estimateur sans biais de $(n - |\xi|)\sigma^2$ et $C_p(\xi)$ vaudra approximativement $|\xi|$. Cette interprétation n'est valable que si le $C_p(\xi)$ est calculé avec d'autres données que celles qui permettent le choix de ξ. A la fin de ce chapitre, une note présente en détail ce problème.
Si nous rajoutons des variables qui n'interviennent pas dans le modèle, la SCR ne va pas beaucoup diminuer mais $|\xi|$ va augmenter, nous aurons alors un $C_p(\xi)$ qui sera plus grand que $|\xi|$.
Si nous avons omis des variables importantes, la SCR sera un estimateur de $(n - |\xi|)\sigma^2$ et d'une quantité positive. Le $C_p(\xi)$ sera donc plus grand que $|\xi|$.

7.3.5 Vraisemblance et pénalisation

Sous l'hypothèse de normalité des résidus, la log-vraisemblance de l'échantillon vaut (section 5.1 p. 91)

$$\begin{aligned}\mathcal{L}(Y, \beta, \sigma^2) &= \log L(Y, \beta, \sigma^2) \\ &= -\frac{n}{2}\log\sigma^2 - \frac{n}{2}\log 2\pi - \frac{1}{2\sigma^2}\|Y - X\beta\|^2. \qquad (7.3)\end{aligned}$$

Le calcul de la log-vraisemblance (évaluée à l'estimateur du maximum de vraisemblance) pour le modèle admettant $|\xi|$ variables vaut alors

$$\mathcal{L}(\xi) = -\frac{n}{2}\log\frac{\mathrm{SCR}(\xi)}{n} - \frac{n}{2}(1+\log 2\pi).$$

Choisir un modèle en maximisant la vraisemblance revient à choisir le modèle ayant la plus petite SCR . Il faut donc introduire une pénalisation. Afin de minimiser un critère, on travaille avec l'opposée de la log-vraisemblance et les critères s'écrivent en général

$$-2\mathcal{L}(\xi) + 2|\xi+1|f(n),$$

où $f(n)$ est une fonction de pénalisation dépendant de n. La pénalisation tient compte non pas du nombre de variables mais du nombre de paramètres à estimer. Dans le cas général, si le modèle admet $|\xi|$ variables, il y a donc $|\xi|$ paramètres à estimer plus l'estimateur de σ^2 d'où $|\xi+1|$.

Akaike Information Criterion (AIC)

Ce critère, introduit par Akaike en 1973, est défini pour un modèle contenant les variables indicées par ξ :

$$\begin{aligned}\mathrm{AIC}(\xi) &= -2\mathcal{L}(\xi) + 2|\xi+1| \\ &= n\log\frac{\mathrm{SCR}(\xi)}{n} + n(1+\log 2\pi) + 2|\xi+1|.\end{aligned}$$

L'AIC est une pénalisation de la log-vraisemblance par deux fois le nombre de paramètres. Nous obtenons une définition équivalente

$$\mathrm{AIC}(\xi) = n(1+\log 2\pi) + n\log\frac{\mathrm{SCR}(\xi)}{n} + 2|\xi+1|.$$

L'utilisation de ce critère est simple : il suffit de le calculer pour tous les modèles ξ concurrents et de choisir celui qui possède l'AIC le plus faible.

Bayesian Information Criterion (BIC)

Le BIC (Schwarz, 1978) est défini comme

$$\mathrm{BIC}(\xi) = -2\mathcal{L}(\xi) + |\xi+1|\log n = n(1+\log 2\pi) + n\log\frac{\mathrm{SCR}(\xi)}{n} + |\xi+1|\log n.$$

L'utilisation de ce critère est identique à celle de l'AIC et nous pouvons constater qu'il revient aussi à pénaliser la log-vraisemblance par le nombre de paramètres $|\xi|$ multiplié par une fonction des observations (et non plus 2). Par définition, $f(n)$ vaut ici $\log n/2$. Ainsi, plus le nombre d'observations n augmente, plus la pénalisation est faible. Cependant, cette pénalisation est en général plus grande que 2 (dès que $n > 7$) et donc le *BIC a tendance à sélectionner des modèles plus petits que l'AIC.*

D'autres critères

A titre d'exemple, Bozdogan (1987) a proposé $2f(n) = \log n + 1$, Hannan & Quinn (1979) ont proposé $f(n) = c \log\log n$ où c est une constante plus grande que 1. Il existe de très nombreuses pénalisations dans la littérature mais les deux les plus répandues sont le BIC et l'AIC.

7.3.6 Liens entre les critères

Avec la procédure de test, nous comparons les modèles emboîtés admettant ξ et $\xi + 1$ variables. Nous conservons le modèle le plus petit, celui à ξ variables, si

$$\delta = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\mathrm{SCR}(\xi+1)/(n - |\xi| - 1)} < 4,$$

où 4 est une approximation pour n grand du fractile $f_{1,n-|\xi|-1}(.95)$. Qu'en est-il des autres critères ?
Commençons par le $\mathrm{R}^2_{\mathrm{a}}$. Si nous choisissons le modèle à $|\xi|$ variables, c'est que nous avons

$$\mathrm{R}^2_{\mathrm{a}}(\xi) > \mathrm{R}^2_{\mathrm{a}}(\xi+1).$$

En récrivant ces termes en fonction des SCR, nous avons

$$\begin{aligned}
\frac{\mathrm{SCR}(\xi)}{n - |\xi|} &< \frac{\mathrm{SCR}(\xi+1)}{n - |\xi+1|} \\
\frac{(n - |\xi| - 1)\,\mathrm{SCR}(\xi)}{\mathrm{SCR}(\xi+1)} &< n - |\xi| - 1 + 1 \\
\delta = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\mathrm{SCR}(\xi+1)/(n - |\xi| - 1)} &< 1.
\end{aligned}$$

Les deux procédures sont parfaitement comparables et il est évident que la procédure avec les tests choisira des modèles de taille inférieure ou égale à la procédure utilisant le $\mathrm{R}^2_{\mathrm{a}}$ ajusté.
De la même façon, analysons le résultat obtenu avec un critère de vraisemblance pénalisée. Si nous choisissons le modèle à ξ variables, nous avons

$$-2\mathcal{L}(\xi) + 2|\xi+1|f(n) \leq -2\mathcal{L}(\xi+1) + 2|\xi+1|f(n) + 2f(n).$$

En remplaçant, nous obtenons

$$\begin{aligned}
\log\frac{\mathrm{SCR}(\xi)}{n} &\leq \log\frac{\mathrm{SCR}(\xi+1)}{n} + 2\frac{f(n)}{n} \\
\mathrm{SCR}(\xi) &\leq \mathrm{SCR}(\xi+1)\exp\frac{2f(n)}{n} \\
\mathrm{SCR}(\xi) &\leq \mathrm{SCR}(\xi+1)\left[\exp\frac{2f(n)}{n} - 1\right] + \mathrm{SCR}(\xi+1).
\end{aligned}$$

Nous obtenons alors

$$\frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\mathrm{SCR}(\xi+1)/(n-|\xi|-1)} \leq (n-|\xi|-1)\left[\exp\frac{2f(n)}{n} - 1\right].$$

Si $2f(n)/n$ est proche de 0, nous approximons la quantité précédente après un développement limité à l'ordre 1 par

$$\delta \quad = \quad \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\mathrm{SCR}(\xi+1)/(n-|\xi|-1)} \leq 2f(n)\left(1 - \frac{|\xi|+1}{n}\right).$$

Le tableau suivant donne les différentes valeurs en fonction du critère utilisé.

Critères classiques	Valeur de δ
Test	4
R^2 ajusté	1
AIC	$2\left(1 - \frac{\lvert\xi\rvert+1}{n}\right)$
BIC	$\log n\left(1 - \frac{\lvert\xi\rvert+1}{n}\right)$

Tableau 7.2 – Valeurs de δ en fonction des critères.

Le $\mathrm{C_p}$ n'est pas intégré dans ce tableau car en écrivant le choix du modèle à ξ variables avec le $\mathrm{C_p}$:

$$\mathrm{C_p}(\xi) < \mathrm{C_p}(\xi+1),$$

nous avons

$$\frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\mathrm{SCR}(p)/(n-p)} \leq 2.$$

Le dénominateur du $\mathrm{C_p}$ est calculé avec toutes les variables initiales et nous n'avons pas le δ des autres procédures. Cependant pour $p << n$, nous pouvons considérer que le δ associé au $\mathrm{C_p}$ vaut environ 2. Le $\mathrm{C_p}$ aura tendance à choisir des modèles plus grands que ceux choisis avec un test entre modèles emboîtés et une erreur de première espèce $\alpha = 5$ %, mais uniquement si l'on choisit comme estimateur de σ^2, la valeur $\mathrm{SCR}(p)/(n-p)$.
En fonction du nombre d'individus n et du nombre de variables sélectionnées, nous pouvons résumer les critères et la taille du modèle dans le tableau suivant :

Critères classiques	Taille $\lvert\xi\rvert$ du modèle
TEST ou BIC	faible
AIC	$\downarrow$
$\mathrm{R_a^2}$	forte

Tableau 7.3 – Comparaison de $|\xi|$ des modèles sélectionnés avec $n > 7$.

Il est délicat d'intégrer le $\mathrm{C_p}(\xi)$ dans ce tableau car lorsque nous avons écrit le $\mathrm{C_p}(\xi)$ sous forme de test, nous avions vu que le dénominateur est calculé avec la $\mathrm{SCR}(p)/(n-p)$. En supposant que les estimateurs de σ^2 (dans un cas $\mathrm{SCR}(p)/(n-p)$ et dans l'autre $\mathrm{SCR}(\xi+1)/(n-|\xi|-1)$) soient presque identiques, la borne du $\mathrm{C_p}(\xi)$ vaut 2 et celle de l'AIC vaut $2(1-(|\xi|+1)/n)$, *l'AIC tend à sélectionner des modèles de taille plus grande que le* $\mathrm{C_p}$.

7.4 Procédure de sélection

La sélection de modèle peut être vue comme la recherche du modèle optimal, au sens d'un critère choisi, parmi toutes les possibilités. Cela peut donc être vu comme une optimisation d'une fonction objectif (le critère). Pour cela, et à l'image des possibilités en optimisation, on peut soit faire une recherche exhaustive car le nombre de modèles possibles est fini, soit partir d'un point de départ et utiliser une méthode d'optimisation de la fonction objectif (recherche pas à pas).
Remarquons qu'en général trouver le minimum global de la fonction objectif n'est pas garanti dans les recherches pas à pas et que seul un optimum local dépendant du point de départ choisi sera trouvé. Si les variables explicatives sont orthogonales, alors l'optimum trouvé sera toujours l'optimum global.

7.4.1 Recherche exhaustive

Effectuer tous les modèles possibles avec p variables donne 2^p-1 possibilités, ce qui n'est pas envisageable si p est grand. Des techniques algorithmiques permettent cependant de minimiser le nombre de calculs à effectuer et permettent d'envisager cette possibilité dans des cas de taille modérée (Miller, 2002).
Remarquons que ce type de recherche n'a aucun sens lorsque l'on souhaite utiliser des tests puisque cette procédure compare uniquement deux modèles emboîtés l'un dans l'autre.

7.4.2 Recherche pas à pas

Ce type de recherche est obligatoire pour les tests puisque l'on ne peut tester que des modèles emboîtés. En revanche, elle ne permet en général que de trouver un optimum local. Il est bon de répéter cette procédure à partir de différents points de départ. Pour les autres critères, ce type de recherche n'est à conseiller que lorsque la recherche exhaustive n'est pas possible (n grand, p grand, etc.).

Méthode ascendante (*forward selection*)

A chaque pas, une variable est ajoutée au modèle.

— Si la méthode ascendante utilise un test F, nous rajoutons la variable X_i dont la probabilité critique (*p-value*) associée à la statistique partielle de test de Fisher qui compare les 2 modèles est minimale. Nous nous arrêtons lorsque toutes les

variables sont intégrées ou lorsque la probabilité critique est plus grande qu'une valeur seuil.

— Si la méthode ascendante utilise un critère de choix, nous ajoutons la variable X_i dont l'ajout au modèle conduit à l'optimisation du critère de choix. Nous nous arrêtons lorsque toutes les variables sont intégrées ou lorsqu'aucune variable ne permet l'optimisation du critère de choix (voir aussi fig. 7.8).

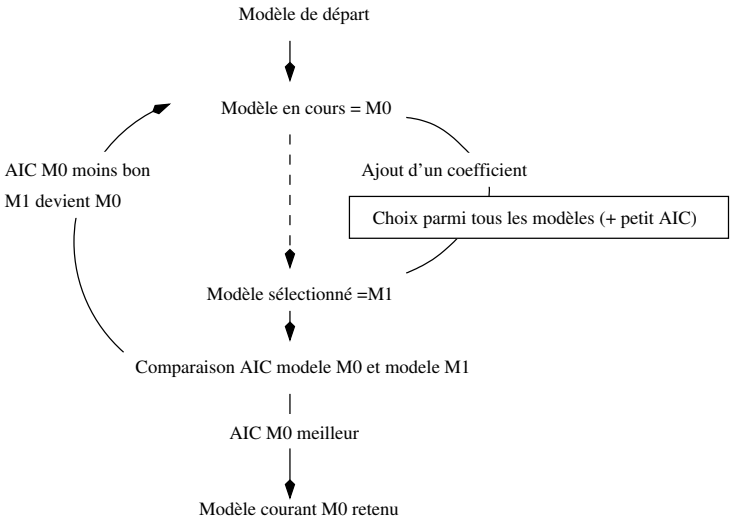


Fig. 7.8 – Technique ascendante utilisant l'AIC.

Méthode descendante (*backward selection*)

A la première étape, toutes les variables sont intégrées au modèle.

— Si la méthode descendante utilise un test F, nous éliminons ensuite la variable X_i dont la *p-value*, associée à la statistique partielle de test de Fisher, est la plus grande. Nous nous arrêtons lorsque toutes les variables sont retirées du modèle ou lorsque la valeur *p-value* est plus petite qu'une valeur seuil.

— Si la méthode descendante utilise un critère de choix, nous retirons la variable X_i qui conduit à l'amélioration la plus grande du critère de choix. Nous nous arrêtons lorsque toutes les variables sont retirées ou lorsqu'aucune variable ne permet l'augmentation du critère de choix.

Méthode progressive (*stepwise selection*)

C'est le même principe que pour la méthode ascendante, sauf que l'on peut éliminer des variables déjà introduites. En effet, il peut arriver que des variables introduites en début ne soient plus significatives après introduction de nouvelles variables.
Remarquons qu'en général la variable « constante », constituée de 1 et associée au coefficient « moyenne générale », est en général traitée à part et elle est toujours présente dans le modèle.

7.5 Exemple : la concentration en ozone

7.5.1 Variables explicatives quantitatives

Nous continuons à analyser le jeu de données de l'ozone et à ne retenir que les variables quantitatives. Le package **leaps** permet d'effectuer une recherche exhaustive lorsque le nombre de variables explicatives n'est pas trop important. Au-delà de 50 variables, l'argument `really.big=TRUE` doit être ajouté, au risque d'un temps de calcul prohibitif. Nous allons donc effectuer cette recherche. Le logiciel propose également de retenir *via* l'argument `nbest`, un nombre défini par l'utilisateur de modèles ayant 1, puis 2, puis 3, ... variables. Nous fixons ce niveau à 1.

```
> library(leaps)
> recherche <- regsubsets(O3 ~ T12 + T15 + Ne12 + N12 + S12 +
+                              E12 + W12 + Vx + O3v, int = T,
+   nbest = 1, nvmax = 10, method = "exhaustive", data = ozone)
```

Pour pouvoir utiliser les résultats de cette procédure, le graphique est l'outil le plus approprié. Le logiciel propose 4 critères de choix : le BIC, le C_p, le R_a^2 et le R^2. Nous allons donc dessiner ces résultats avec les 4 critères. Le graphique associé retourne en abscisse les variables et pour une ligne donnée, les variables qui sont sélectionnées. La ligne du haut correspond au meilleur modèle et il faut donc regarder les variables cochées qui correspondent aux variables sélectionnées.

Minimisation du BIC

```
> plot(recherche, scale = "bic")
```

Nous obtenons le graphique suivant :

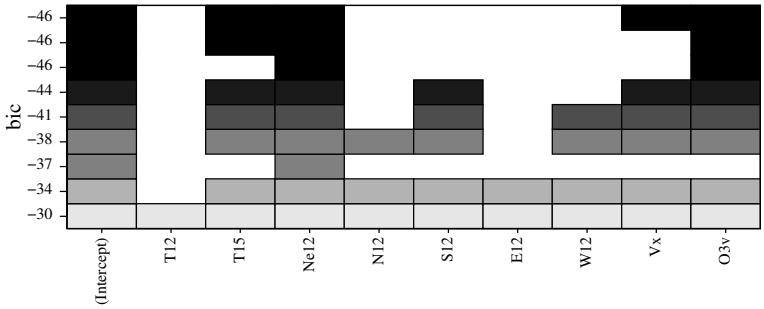


Fig. 7.9 – Méthode exhaustive, critère du BIC.

Le modèle retenu alors serait le modèle à 5 variables

$$\texttt{O3} = \beta_1 + \beta_2\texttt{T15} + \beta_3\texttt{Ne12} + \beta_4\texttt{Vx} + \beta_5\texttt{O3v} + \varepsilon.$$

Minimisation du $\mathrm{C_p}$

```
> plot(recherche, scale = "Cp")
```

Nous obtenons le graphique suivant :

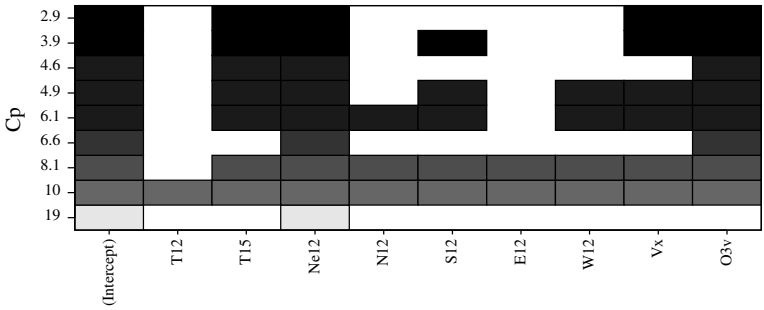


Fig. 7.10 – Méthode exhaustive, critère du $\mathrm{C_p}$.

Le modèle retenu est identique au modèle retenu par le critère du BIC.

Maximisation du $\mathrm{R_a^2}$

```
> plot(recherche, scale = "adjr2")
```

Nous obtenons le graphique suivant :

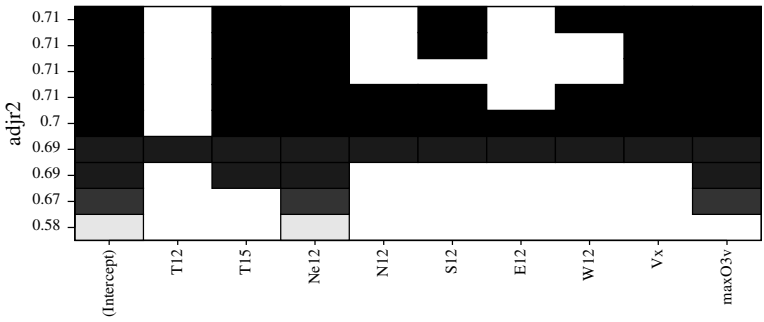


Fig. 7.11 – Méthode exhaustive, critère du $\mathrm{R_a^2}$.

Le modèle retenu admet plus de variables que les modèles retenus avec les critères précédents. Nous avons

$$\texttt{O3} = \beta_1 + \beta_2\texttt{T15} + \beta_3\texttt{Ne12} + \beta_4\texttt{S12} + \beta_5\texttt{W12} + \beta_6\texttt{Vx} + \beta_7\texttt{O3v} + \varepsilon.$$

Maximisation du R^2

```
> plot(recherche, scale = "r2")
```

Nous obtenons le graphique suivant :

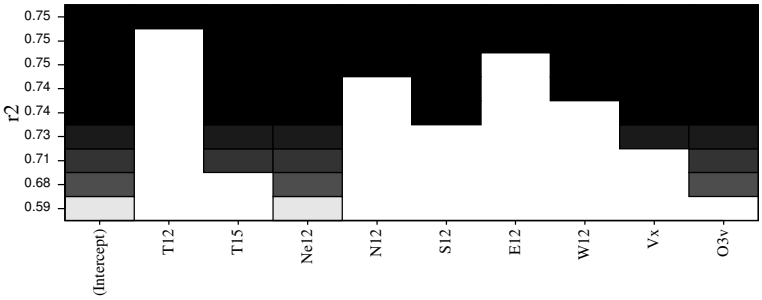


Fig. 7.12 – Méthode exhaustive, critère du R^2 .

Comme prévu, nous conservons avec ce critère toutes les variables initiales.

Il est intéressant de pouvoir retrouver le modèle choisi dans la sortie de la fonction **regsubsets**. Comme nous venons de le voir le modèle trouvé par l'algorithme dépend du critère choisi par l'utilisateur. L'objet R `recherche` est une liste et pour pouvoir extraire le modèle sélectionné, il faut utiliser les ordres suivants :

```
> resume <- summary(recherche)
> nomselec <- colnames(resume$which)[
                        resume$which[which.min(resume$bic),] ][-1]
```

Le premier nom correspondant à `(Intercept)` nous l'éliminons grâce à `[-1]`. Avec ces noms des variables sélectionnées, nous construisons une formule décrivant notre modèle sélectionné :

```
> formule <- formula(paste("O3~",paste(nomselec,collapse="+")))
```

Enfin, nous ajustons notre modèle sélectionné par BIC avec la fonction **lm**, la formule et les données :

```
> modeleslectionne <- lm(formule, data = ozone)
```

Bien évidemment si l'utilisateur souhaite choisir un autre critère comme le C_p (il faut choisir le minimum du critère) ou le R_a^2 (il faut alors choisir le maximum du critère et remplacer **which.min** par **which.max**.

7.5.2 Intégration de variables qualitatives

Il arrive fréquemment que, parmi les variables explicatives, certaines soient qualitatives. Ce problème a largement été abordé dans le chapitre 6 où nous avons vu que la solution naïve consiste à partir d'un codage disjonctif complet : chaque modalité est transformée en un vecteur d'indicatrices d'appartenance à la modalité. Cependant, un tel codage revient à utiliser un coefficient pour chaque modalité de la variable dans le modèle et conduit à une surparamétrisation du modèle qui n'est alors plus identifiable. L'utilisateur (ou le logiciel) choisit en général une contrainte pour s'assurer de l'identifiabilité des coefficients. Pour le langage R, il s'agit de la contrainte `treatment` : la première modalité (dans l'ordre d'apparition de la fonction `levels`) est le traitement de référence et les coefficients des facteurs sont simplement l'écart à cette modalité.

Le package **leaps** (par défaut) va donc transformer chaque variable qualitative en $I - 1$ variables indicatrices. Pour la variable qualitative `vent` (avec $I = 4$ modalités), nous aurons donc trois variables indicatrices : `OUEST`, `NORD` et `SUD` (la modalité `EST` est le niveau de base). La variable `nebulosite` (avec $I = 2$ modalités) donnera donc une seule variable `SOLEIL` (la modalité `NUAGEUX` est le niveau de base). Ensuite le package **leaps** considère chaque colonne comme une variable à part entière et donc les variables indicatrices sont traitées séparément. Cela veut dire que la fonction **regsubsets** peut sélectionner un sous-ensemble de modalité parmi toutes les modalités d'une variable qualitative.

Reprenons l'exemple de l'ozone précédent où nous intégrons les deux variables qualitatives `vent` qui admet 4 modalités (EST...) et `nebulosite` qui en admet 2

```
> library(leaps)
> recherche.ex.qual <- regsubsets(O3 ~ ., int = T, nbest = 1,
          nvmax = 10, method = "exhaustive", data = ozone)
```

Nous voyons en analysant les coefficients du meilleur modèle retenu par exemple avec le critère du BIC

```
> coef(recherche.ex.qual,which.min(summary(recherche.ex)$bic))
(Intercept)         T15        Ne12         O3v   ventOUEST
 64.0195557   1.1177149  -4.0435737   0.2622327  -9.5269595
```

que la seule modalité retenue est la modalité `OUEST`. Cela veut dire que la fonction **regsubsets** a considéré que l'écart entre les 3 autres modalités (EST, NORD, SUD) n'est pas suffisant pour justifier d'inclure les indicatrices pour le vent du NORD et du SUD. Le modèle calculé ici, outre les variables quantitatives retenues, est donc une constante avec une seule indicatrice `ventOUEST`. Le coefficient négatif (qui vaut -9.53) montre que lorsque le vent est à l'OUEST l'ozone chute (coefficient négatif). Par contre pour les autres directions, rien n'est changé, on reste à la constante.

Afin de retrouver le modèle final, il faut procéder « à la main » car la variable `ventOUEST` n'existe pas dans le data-frame initial. Pour cela, nous sélectionnons les noms des variables du meilleur modèle grâce à la procédure évoquée plus haut.

```
> resume <- summary(recherche.ex.qual)
> nomselec <- colnames(resume$which)[
                        resume$which[which.min(resume$bic),] ][-1]
```

Ensuite nous codons les variables qualitatives dans un tableau complet (via la fonction `model.matrix` qui prend les mêmes arguments que `lm` mais qui renvoie seulement la matrice X). De ce tableau complet qui contient donc les variables quantitatives et les indicatrices des modalités (et surtout celle sélectionnée : `ventOUEST`), nous gardons celles souhaitées et nous ajoutons la variable à expliquer. Enfin nous refaisons le modèle sur ce tableau qui ne contient que les variables utiles.

```
> Xcomplet <- model.matrix(O3 ~ ., data = ozone)
> df <- cbind(ozone[, "O3"], Xcomplet[, nomselec])
> modelebic <- lm(O3 ~ ., data = df)
```

La question de sélectionner des modalités dans le choix de variables est une question qui n'admet pas une réponse simple. Il est souvent important de conserver toutes les modalités d'une variable qualitative. La fonction **step** propose d'enlever (ou de conserver chaque variable qualitative) dans sa globalité en tenant compte dans le calcul de l'AIC du nombre de modalités.

```
> choix.step <- step(lm(O3~.,data=ozone))
Start:  AIC=269.88
O3 ~ T12 + T15 + Ne12 + N12 + S12 + E12 + W12 + Vx + O3v +
     nebulosite + vent
...........
```

La fonction part du modèle complet calcule l'AIC en enlevant chaque variable et choisit d'enlever la variable qui une fois enlevée fait le plus baisser l'AIC et itère. L'algorithme s'arrête lorsque l'AIC ne baisse plus. Le modèle final retenu se trouve dans l'objet `choix.step` et il est possible de faire un **predict** directement sur cet objet.

7.6 Exercices

Exercice 7.1 (Questions de cours)

1) Un modèle à p variables a été estimé donnant un R^2 noté $\mathrm{R}^2(p)$. Une nouvelle variable explicative est rajoutée au modèle précédent, après estimation un nouvel R^2 noté $\mathrm{R}^2(p+1)$ est obtenu.
   A. $\mathrm{R}^2(p+1)$ est toujours plus grand que $\mathrm{R}^2(p)$,
   B. $\mathrm{R}^2(p+1)$ est parfois plus petit parfois plus grand, cela dépend de la variable rajoutée,
   C. $\mathrm{R}^2(p+1)$ est toujours plus petit que $\mathrm{R}^2(p)$.

2) Le R^2 permet-il de sélectionner des modèles ?
A. jamais,
B. toujours,
C. oui si les modèles admettent le même nombre de variables explicatives.
3) Vous travaillez avec un modèle restreint ξ par rapport au vrai modèle (des variables sont omises), l'estimateur $\hat{\beta}_\xi$ de β_ξ dans ce nouveau modèle est :
A. toujours biaisé,
B. parfois biaisé,
C. jamais biaisé.

Exercice 7.2 (Analyse du biais)
Démontrer la proposition 7.1 p. 162.

Exercice 7.3 († Variance des estimateurs)
Démontrer la proposition 7.2 p. 163.

Exercice 7.4 (Choix de variables)
Nous considérons le modèle de régression multiple avec 4 variables explicatives. Nous avons fait toutes les régressions possibles.
Les résultats numériques avec $n = 10$ et entre parenthèses la valeur absolue de la statistique de test se trouvent dans le tableau ci-dessous.
Prenez pour fractile de la loi de Student (ddl < 10) la valeur 2.3.

	modèle	R^2	AIC	BIC
M1	$\hat{Y} = -1.24_{(3.3)} + 0.12_{(41.9)}x_1$	.996	-2.18	-2.12
M2	$\hat{Y} = 2.11_{(2.6)} + 0.33_{(15.3)}x_2$	.967	-0.20	-0.14
M3	$\hat{Y} = -38.51_{(9.2)} + 0.52_{(12.5)}x_3$	.952	0.18	0.24
M4	$\hat{Y} = -53.65_{(14.8)} + 0.66_{(18.6)}x_4$	.977	-0.58	-0.52
M12	$\hat{Y} = -1.59_{(2.6)} + 0.13_{(6.9)}x_1 - 0.04_{(0.7)}x_2$	.996	-2.06	-1.97
M13	$\hat{Y} = 1.40_{(0.3)} + 0.12_{(8.4)}x_1 - 0.04_{(0.5)}x_3$	.996	-2.03	-1.94
M14	$\hat{Y} = -8.37_{(1.0)} + 0.10_{(5.6)}x_1 + 0.09_{(0.9)}x_4$	.996	-2.09	-2.00
M23	$\hat{Y} = -13.29_{(1.3)} + 0.21_{(2.6)}x_2 + 0.19_{(1.5)}x_3$	.975	-0.27	-0.18
M24	$\hat{Y} = -31.2_{(3.2)} + 0.14_{(2.4)}x_2 + 0.39_{(3.5)}x_4$	.988	-0.99	-0.90
M34	$\hat{Y} = -58_{(8.21)} - 0.16_{(0.7)}x_3 + 0.87_{(3)}x_4$	.979	-0.46	-0.37
M123	$\hat{Y} = 0.95_{(0.2)} + 0.14_{(5.6)}x_1 - 0.04_{(0.7)}x_2 - 0.03_{(0.5)}x_3$	.996	-1.90	-1.78
M124	$\hat{Y} = -7.4_{(0.8)} + 0.11_{(3.5)}x_1 - 0.03_{(0.5)}x_2 + 0.07_{(0.6)}x_4$	.996	-1.93	-1.80
M134	$\hat{Y} = -12.7_{(1.9)} + 0.1_{(7.5)}x_1 - 0.19_{(2.5)}x_3 + 0.31_{(2.6)}x_4$	.998	-2.59	-2.47
M234	$\hat{Y} = -34.9_{(4.2)} + 0.16_{(3.3)}x_2 - 0.3_{(2)}x_3 - 0.7_{(3.8)}x_4$	.993	-1.30	-1.18
M1234	$\hat{Y} = -13.5_{(1.8)} + 0.1_{(3.7)}x_1 + 0.02_{(0.3)}x_2$ $-0.2_{(2.2)}x_3 + 0.34_{(2.3)}x_4$	.998	-2.40	-2.25

1) Choisissez un modèle en vous basant sur les critères AIC et BIC.
2) Faire de même en utilisant le $\mathrm{R}^2_{\mathrm{a}}$ et des procédures de tests (vous prendrez comme fractile de la loi à Student 2.3).

Exercice 7.5 (Utilisation du $\mathbf{R}^2$)
Soit $Z_{(n,q)}$ une matrice de rang q et soit $X_{(n,p)}$ une matrice de rang p composée des q vecteurs de Z et de $p - q$ autres vecteurs linéairement indépendants. Nous avons les deux

modèles suivants :

$$\begin{aligned} Y &= Z\beta + \varepsilon \\ Y &= X\beta^* + \eta. \end{aligned}$$

Comparer les R^2 dans les deux modèles. Discuter de l'utilisation du R^2 pour le choix de variables.

Exercice 7.6 (Cas orthogonal)
Nous allons supposer que les variables explicatives sont orthogonales et de norme unité. La matrice X est donc une matrice orthogonale et $X'X = I_p$. Nous supposerons également σ^2 connu.

1) Montrer que l'estimateur des moindres carrés s'écrit

$$\hat{\beta} = X'Y = \beta + X'\varepsilon.$$

2) Montrer que la somme des résidus vaut

$$\mathrm{SCR} = \|Y - X\hat{\beta}\|^2 = \sum_{i=1}^{n} y_i^2 - \sum_{j=1}^{p} \hat{\beta}_j^2. \tag{7.4}$$

3) Supposons que nous estimons un modèle ξ. Montrer, en utilisant l'orthogonalité des colonnes de X, que le vecteur des estimateurs obtenus par MCO avec ce modèle (noté $\hat{\beta}_\xi$) sont les coordonnées ξ du vecteur des estimateurs obtenus par MCO $\hat{\beta}$ obtenu avec le modèle complet à p variables.

4) Supposons que le passage du modèle ξ au modèle $\xi + 1$ consiste à ajouter la variable l. Montrer, en utilisant (7.4), que quand nous utilisons les critères de choix de modèle basés sur la pénalisation de la vraisemblance (AIC, BIC...), nous choisissons le modèle $\xi + 1$ (ou nous ajoutons la variable l) si

$$\frac{\hat{\beta}_l^2}{\sigma^2} > 2f(n)$$

5) Montrer que $N = \frac{\mathrm{SCR}(\xi) - \mathrm{SCR}(\xi+1)}{\sigma^2}$ suit une loi $\chi^2(1)$. Déduire (avec l'aide de R) que le test basé sur la statistique N conduit à ajouter la variable l (avec une erreur de première espèce de $\alpha = 5\%$) si

$$\frac{\hat{\beta}_l^2}{\sigma^2} > 3.84$$

6) En utilisant l'approximation suivante $\frac{\mathrm{SCR}(\xi)}{n - |xi|} \approx \sigma^2$, montrer qu'utiliser le R_a^2 pour savoir si nous ajoutons la variable l revient regarder si on a bien (approximativement)

$$\frac{\hat{\beta}_l^2}{\sigma^2} > 1$$

7) En utilisant le fait que σ^2 est connu, nous avons que

$$\mathrm{C_p}(\xi) = \frac{\mathrm{SCR}(\xi)}{\sigma^2} - n + 2|\xi|. \tag{7.5}$$

Montrer qu'utiliser le $\mathrm{C_p}$ pour savoir si nous ajoutons la variable l revient à regarder si on a

$$\frac{\hat{\beta}_l^2}{\sigma^2} > 2$$

8) Construire un algorithme simple (n'utilisant que l'estimation par MCO dans le modèle complet à p variables) afin de sélectionner les variables avec l'un des critères précédents.

7.7 Note : $\mathrm{C_p}$ et biais de sélection

Dans la section consacrée au $\mathrm{C_p}(\xi)$, nous avons insisté sur le caractère non aléatoire de P_{X_ξ} et évoqué le problème de **biais de sélection**. L'objectif de cette note est, au travers d'un exemple simple, de mettre en évidence cette notion.
Soit $X_1, X_2, \ldots, X_p$ des variables orthogonales de norme unité. La matrice X est donc une matrice orthogonale et $X'X = I_p$. L'estimateur des moindres carrés s'écrit alors

$$\hat{\beta} \quad = \quad (X'X)^{-1}X'Y = X'(X\beta + \varepsilon) = \beta + X'\varepsilon.$$

Si l'hypothèse de normalité des résidus est vérifiée, alors $X'\varepsilon$ suit une loi normale de moyenne nulle et de variance $\sigma^2 I_n$. Nous avons alors $\hat{\beta} \sim \mathcal{N}(\beta, \sigma^2 I_n)$. Pour reformuler le $\mathrm{C_p}$ nous devons nous intéresser à la valeur de $\mathrm{SCR}(\xi)$:

$$\begin{aligned}
\mathrm{SCR}(\xi) &= \|Y - \hat{Y}_\xi\|^2 = \|P_{X^\perp}Y + P_X Y - P_{X_\xi}Y\|^2 \\
&= \|P_{X^\perp}Y\|^2 + \|P_X Y - P_{X_\xi}Y\|^2 = (n-p)\hat{\sigma}^2 + \|P_X Y - P_{X_\xi}Y\|^2 \\
&= (n-p)\hat{\sigma}^2 + \|P_{X_\xi^\perp}P_X Y + P_{X_\xi}P_X Y - P_{X_\xi}Y\|^2.
\end{aligned}$$

Notons $\bar{\xi}$ l'ensemble des indices des variables non incluses dans le modèle ξ (le complémentaire par rapport à $\{1, 2, \ldots, p\}$), nous avons, en nous rappelant que $P_X Y = X\hat{\beta}$ et que toutes les variables sont orthogonales :

$$\begin{aligned}
\mathrm{SCR}(\xi) &= (n-p)\hat{\sigma}^2 + \|P_{X_\xi^\perp}X\hat{\beta} + P_{X_\xi}Y - P_{X_\xi}Y\|^2 = (n-p)\hat{\sigma}^2 + \|X_{\bar{\xi}}\hat{\beta}_{\bar{\xi}}\|^2 \\
&= (n-p)\hat{\sigma}^2 + \hat{\beta}'_{\bar{\xi}} X_{\bar{\xi}}{}' X_{\bar{\xi}} \hat{\beta}_{\bar{\xi}} \\
&= (n-p)\hat{\sigma}^2 + \sum_{j \notin \xi} \hat{\beta}_j^2. \qquad (7.6)
\end{aligned}$$

La définition du $\mathrm{C_p}(\xi)$ (équation 7.2) donne

$$\hat{\sigma}^2\,\mathrm{C_p}(\xi) = \mathrm{SCR}(\xi) - (n - 2|\xi|)\hat{\sigma}^2.$$

En remplaçant dans cette équation la quantité $\mathrm{SCR}(\xi)$, nous avons

$$\begin{aligned}
\hat{\sigma}^2\,\mathrm{C_p}(\xi) &= (n-p)\hat{\sigma}^2 + \sum_{j \notin \xi} \hat{\beta}_j^2 - (n - 2|\xi|)\hat{\sigma}^2 \\
&= \sum_{j \notin \xi} \hat{\beta}_j^2 + (2|\xi| - p)\hat{\sigma}^2 \\
&= \sum_{j=1}^{p} \hat{\beta}_j^2 - \sum_{j \in \xi} \hat{\beta}_j^2 - p\hat{\sigma}^2 + 2|\xi|\hat{\sigma}^2.
\end{aligned}$$

Nous avons $p\hat{\sigma}^2$ que nous mettons dans la première somme de p termes et $2|\xi|\hat{\sigma}^2$ que nous mettons dans la seconde somme de $|\xi|$ termes. Cela donne

$$\hat{\sigma}^2\,\mathrm{C_p}(\xi) \quad = \quad \sum_{j=1}^{p} (\hat{\beta}_j^2 - \hat{\sigma}^2) - \sum_{j \in \xi} (\hat{\beta}_j^2 - 2\hat{\sigma}^2).$$

Choix de variables, $|\xi|$ fixé

Si nous souhaitons, grâce au critère du $\hat{\sigma}^2\,\mathrm{C_p}$, sélectionner parmi les ensembles ξ de cardinal $|\xi|$ fixé, nous allons donc devoir chercher l'ensemble de $\mathrm{SCR}(\xi)$ minimum, soit celui dont les normes $\hat{\beta}_j^2, j \in \xi$, sont maximales (ou minimales dans le complémentaire). La procédure conduit donc à sélectionner les $|\xi|$ variables dont les coefficients estimés sont les plus grands en valeur absolue.

Choix de variables, $|\xi|$ non fixé

Maintenant, nous considérons que le cardinal $|\xi|$ est variable. Si ce cardinal est 1, alors nous choisissons la variable dont le coefficient estimé est le plus grand et le $\mathrm{C_p}(1)$ vaut

$$\mathrm{C_p}(1) = \sum_{j=1}^{p}(\hat{\beta}_j^2 - \hat{\sigma}^2) - (\hat{\beta}_{(1)}^2 - 2\hat{\sigma}^2),$$

où $\hat{\beta}_{(1)}$ est le coefficient associé à la variable admettant la plus grande valeur des $\hat{\beta}_i$. Maintenant, comme le $|\xi|$ est variable, nous souhaitons savoir si des ensembles ξ de cardinal 2 conduisent à une diminution du $\mathrm{C_p}$. Nous savons que l'une des deux variables est la même que celle sélectionnée quand $|\xi| = 1$. La deuxième variable est ajoutée au modèle optimal de cardinal $|\xi| = 1$. Si l'ajout de cette variable permet une diminution du $\hat{\sigma}^2\,\mathrm{C_p}$, alors le modèle optimum de cardinal 2 est préféré à celui de cardinal 1. Le $\mathrm{C_p}(2)$ vaut

$$\mathrm{C_p}(2) = \sum_{j=1}^{p}(\hat{\beta}_j^2 - \hat{\sigma}^2) - (\hat{\beta}_{(1)}^2 - 2\hat{\sigma}^2) - (\hat{\beta}_{(2)}^2 - 2\hat{\sigma}^2).$$

La différence des $\mathrm{C_p}$ vaut

$$\Delta_{1-2} = \mathrm{C_p}(1) - \mathrm{C_p}(2) = \hat{\beta}_{(2)}^2 - 2\hat{\sigma}^2.$$

Si $\Delta_{1-2} > 0$, c'est-à-dire $\hat{\beta}_{(2)}^2 > 2\hat{\sigma}^2$, alors le modèle de cardinal 2 est préféré à celui de cardinal 1. De même pour le passage du cardinal 2 à celui du cardinal 3 ; à chaque fois la différence de $\hat{\sigma}^2\,\mathrm{C_p}$ diminue car par définition $\hat{\beta}_{(j)}^2$ diminue quand j augmente. *Au final, le modèle retenu sera celui dont les carrés des coefficients estimés sont tous plus grands que* $2\hat{\sigma}^2$.

Espérance du $\mathrm{C_p}$

Si maintenant nous nous intéressons à ce que donne cette sélection en moyenne, calculons l'espérance de $\hat{\sigma}^2\,\mathrm{C_p}$. Simplifions les calculs en supposant tous les β_j nuls. Nous savons que $\mathbb{E}(\hat{\beta}_j^2) = \beta_j^2 + \sigma^2 = \sigma^2$ et que $\hat{\sigma}^2$ est un estimateur sans biais de σ^2. Le premier terme $\sum_{j=1}^{p}(\hat{\beta}_j^2 - \hat{\sigma}^2)$ a une espérance nulle. Si nous nous intéressons au second terme $\sum_{j\in\xi}(\hat{\beta}_j^2 - 2\hat{\sigma}^2)$, nous savons que tous les $\hat{\beta}_j$ sélectionnés dans ξ sont tels que $\hat{\beta}_j^2 > 2\hat{\sigma}^2$, donc ce terme est toujours positif et donc son espérance aussi. En conclusion, $\hat{\sigma}^2\,\mathrm{C_p}$ est donc en moyenne négatif, alors qu'il est censé donner une idée de la qualité d'approximation *via* l'EQM, qui est une quantité positive ! *Le* $\mathrm{C_p}$ *va donc sous-estimer en moyenne l'EQM, il sera trop optimiste.*

Espérance de la taille du modèle $|\xi|$

Analysons en moyenne la dimension du modèle sélectionné par $\mathrm{C_p}$. La taille $|\xi|$ est le nombre de coefficients $\hat{\beta}_j$ qui sont tels que $\hat{\beta}_j^2 > 2\hat{\sigma}^2$, ce qui s'écrit :

$$\begin{aligned}\mathbb{E}(|\xi|) &= \sum_{j=1}^{p} \mathbb{E}(\mathbb{1}_{\{\hat{\beta}_j^2 > 2\hat{\sigma}^2\}}) = \sum_{j=1}^{p} \mathbb{E}(\mathbb{1}_{\{\hat{\beta}_j^2/\hat{\sigma}^2 > 2\}}) \\ &= p\,\Pr(\hat{\beta}_j^2/\hat{\sigma}^2 > 2) = p\,\Pr(|\hat{\beta}_j/\hat{\sigma}| > \sqrt{2}) = 2p\,\Pr(\hat{\beta}_j/\hat{\sigma} > \sqrt{2}),\end{aligned}$$

avec $\Pr(.)$ dénotant la probabilité. Or $\hat{\beta}_j \sim \mathcal{N}(0, \sigma^2)$, $(n-p)\hat{\sigma}^2/\sigma^2 \sim \chi^2(n-p)$ et ces deux variables aléatoires sont indépendantes, donc $\sigma\hat{\beta}_j/\hat{\sigma} \sim t(n-p)$ et donc $\mathbb{E}(|\xi|) = 2p\ t_{\sqrt{2}\sigma}(n-p) > 0$. Si nous supposons pour fixer les idées que $\hat{\sigma}^2 = \sigma^2$, nous avons alors une loi normale centrée réduite et $\mathbb{E}(|\xi|) = 2p\ z_{\sqrt{2}} \approx 0.16p$. Rappelons que tous les coefficients β_j sont supposés égaux à 0 et donc que la taille $|\xi|$ idéale est 0. *La taille sélectionnée est donc en moyenne toujours trop grande.*

Conclusion

Le $\mathrm{C_p}$, quand il est utilisé de manière classique sur le même jeu de données que celui utilisé pour estimer les paramètres, conduit à sélectionner les variables associées à de grandes valeurs de paramètres. Lorsque l'on considère la moyenne sur tous les échantillons sur lesquels on applique la procédure de sélection/estimation, les variables sélectionnées seront celles qui auront des valeurs élevées pour leur coefficient. Si l'on applique la même procédure d'estimation, suivie de la sélection du modèle par Cp, alors en moyenne cela conduit à des modèles dont les coefficients sont trop élevés en valeur absolue. Certains cas de figure vont être exclus par la procédure de sélection. Nous ne pourrons jamais obtenir de modèle avec la variable 1 sélectionnée quand le coefficient estimé est inférieur à celui de la variable 2 (fig. 7.13). L'exclusion de ces cas conduit à des coefficients biaisés vers de plus grandes valeurs absolues. Ce biais est quelquefois appelé biais de sélection (Miller, 2002).

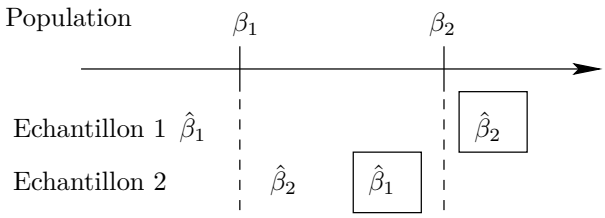


Fig. 7.13 – Biais de sélection dans un modèle à une variable sélectionnée. Le coefficient encadré est celui de la variable sélectionnée.

Ces conclusions sont valides dans le cas où les variables sont orthogonales. Pour généraliser ces résultats, l'équation (7.6) devient $(n-p)\hat{\sigma}^2 + \|P_{X_\xi}^{\perp} X\hat{\beta}\|^2$, ce qui conduit, avec la définition de $\hat{\sigma}^2\,\mathrm{C_p}$, à l'équation suivante :

$$\hat{\sigma}^2\,\mathrm{C_p} = \|P_{X_\xi}^{\perp} X\hat{\beta}\|^2 + (2|\xi| - p)\hat{\sigma}^2.$$

Ici la matrice X n'est pas orthogonale, donc les normes des variables explicatives ne sont pas toutes identiques et égales à 1, en d'autres termes les échelles (ou dispersions) sont différentes. De plus, les variables explicatives sont peut-être corrélées. La sélection par le $\mathrm{C_p}$ va donc favoriser les variables qui mènent à un terme $\|P^{\perp}_{X_\xi} X\hat{\beta}\|^2$ faible. Ceci dépend donc de la valeur des coefficients estimés, de la norme de la variable mais aussi des corrélations qu'une variable entretient avec les autres variables. Ainsi prenons le cas où toutes les variables explicatives ont la même norme et deux variables, numérotées par exemple 3 et 4, sont très fortement corrélées. Si l'on en prend une, par exemple la 3, dans l'ensemble ξ, alors pour la seconde, même si son coefficient $\hat{\beta}_4$ est élevé par rapport aux autres, la projection dans l'orthogonal de $\Im(X_\xi)$ de $X_4\hat{\beta}_4$ sera de norme peu élevée puisque X_3 et X_4 sont très corrélées. Ainsi la variable 4 ne sera pas forcément sélectionnée.

Chapitre 8

Régularisation des moindres carrés : ridge, lasso, elastic-net

8.1 Introduction

Dans ce chapitre, nous nous intéressons au problème de régression sous contrainte de norme. Rappelons que le problème initial des moindres carrés consiste à chercher le (vecteur de) coefficient(s) $\hat{\beta}$ qui minimise le critère des moindres carrés :

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|Y - X\beta\|^2. \tag{8.1}$$

Afin de pouvoir résoudre ce problème, nous avons introduit l'hypothèse $\mathcal{H}_1$: la matrice X est de plein rang. Cette hypothèse permet alors de trouver une solution unique au problème posé, à savoir $\hat{\beta} = (X'X)^{-1}X'Y$.
Dans certains cas, le problème est mal posé et non résolvable en l'état. Par exemple, si $p > n$ alors il y a plus de coefficients à estimer que d'observations et la matrice X n'est plus de plein rang (hypothèse $\mathcal{H}_1$ non vérifiée). Dans ce cas, l'inverse $(X'X)^{-1}$ n'existe plus et il n'existe plus de solution unique au problème des MCO (8.1). On est confronté au même genre de problème lorsqu'au moins une variable explicative est combinaison linéaire des autres :

$$\exists j \quad : \quad X_j = \sum_{i \neq j} \alpha_i X_i, \ \alpha_i \in \mathbb{R}$$

Nous avons vu dans la section 4.1 qu'une solution à ce problème était d'augmenter artificiellement la diagonale de $X'X$ pour obtenir l'estimateur ridge

$$\hat{\beta}_{\text{ridge}}(\lambda) = (X'X + \lambda I)^{-1}X'Y$$

où $\lambda > 0$ est un paramètre à sélectionner. Nous avons également montré que l'estimateur ridge est solution du problème de minimisation du critère des moindres carrés pénalisé par la norme de β :

$$\hat{\beta}_{\text{ridge}}(\lambda) = \underset{\beta \in \mathbb{R}^p}{\text{argmin}} \left\{ \|Y - X\beta\|^2 + \lambda \|\beta\|^2 \right\}.$$

Il est bien entendu possible d'utiliser d'autres termes pour régulariser le critère des moindres carrés. D'une manière générale, étant donné $J(.)$ une fonction de régularisation convexe à valeurs positives et λ un réel positif, on cherche à minimiser

$$\|Y - X\beta\|^2 + \lambda J(\beta). \tag{8.2}$$

Le paramètre λ assure la balance entre la régularité de la solution (faible valeur de $J(\beta)$) et son adéquation au problème initial posé (faible valeur de $\|Y - X\beta\|^2$). Le choix de λ et de la fonction de régularisation $J(.)$ vont bien entendu se révéler cruciaux pour les performances des estimateurs. Comment choisir la fonction de régularisation et comment choisir λ ?
Tout d'abord il est important de rappeler ici que le problème de minimisation des MCO pénalisés (problème dual) :

$$\min_{\beta \in \mathbb{R}^p} \|Y - X\beta\|^2 + \lambda J(\beta)$$

est équivalent par dualité [1] au problème de minimisation des MCO sous contraintes (problème primal) :

$$\min_{\beta \in \mathbb{R}^p : J(\beta) \leq \delta} \|Y - X\beta\|^2.$$

Ainsi l'une ou l'autre des formulations est équivalente. Illustrons maintenant graphiquement le choix de la fonction $J(.)$ via la formulation « MCO sous contraintes ». Choisissons un modèle admettant deux variables explicatives X_1 et X_2 et donc deux coefficients β_1 et β_2 :

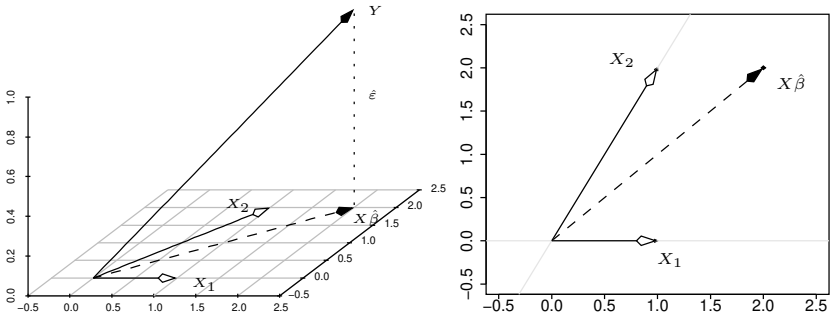
$$Y = X\beta + \varepsilon = X_1\beta_1 + X_2\beta_2 + \varepsilon.$$

Les données figurant dans le tableau 8.1 permettent de calculer $\hat{\beta}$ qui vaut ici $(1, 1)'$ et nous pouvons obtenir aussi la représentation graphique de la figure 8.1a. Le plan de $\Im(X)$ (vu de dessus) est donné dans la figure 8.1b.

Y	X_1	X_2
2	1	1
2	0	2
1	0	0

Tableau 8.1 – Données pour l'illustration graphique 3D.

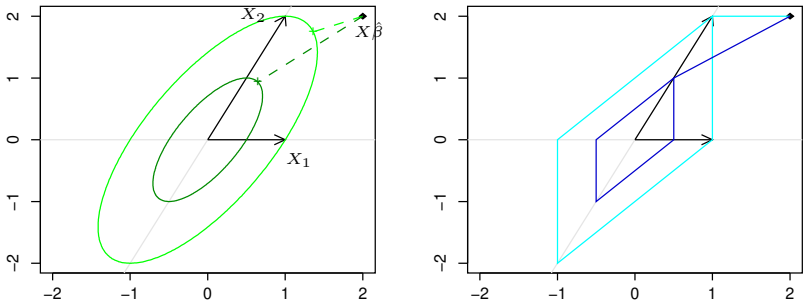
1. Le problème primal est réalisable, avec une valeur finie ; $J() \leq \delta$ est un sous-ensemble convexe de $\mathbb{R}^p$ d'intérieur non vide ; sur ce sous-ensemble, les fonctions $\beta \mapsto \|Y - X\beta\|^2$ et $J()$ sont convexes (voir Bertsekas 2016, p. 589).



(a) Représentation de Y, $P_X Y = X\hat{\beta}$ et $\hat{\varepsilon}$ **(b)** Représentation du plan $\Im(X)$.

Fig. 8.1 – Représentation graphique des données du tableau 8.1.

Nous allons considérer pour la fonction $J(\beta)$ une norme l^2 au carré (*i.e.* $J(\beta) = \|\beta\|^2 = \sum_{i=1}^p \beta_i^2$) qui correspond à la régression ridge et une norme l^1 (*i.e.* $J(\beta) = \|\beta\|_1 = \sum_{i=1}^p |\beta_i|$) qui correspond à la régression lasso. Représentons ces deux possibilités pour $J(\beta)$, avec une norme de 1 et de 0.5 dans chaque cas (donc 1 et 0.25 pour la norme l^2 au carré) dans le sous-espace $\Im(X)$. Les contraintes sur β engendrent un sous-espace $\Im(X)$ contraint. Dans cet exemple nous considérerons que $\hat{\beta}$ des MCO ne satisfait pas la contrainte sinon bien évidemment la solution du problème est $\hat{\beta}$!



(a) $J(\beta)$ est une norme l^2 au carré **(b)** $J(\beta)$ est une norme l^1

Fig. 8.2 – Représentation du sous-espace $\Im(X)$ avec contraintes de norme .5 et 1.

Dans le cas de la régression ridge (fig. 8.2a) nous voyons que parmi les points qui satisfont la contrainte (*i.e.* tous les $(\beta_1, \beta_2)'$ à l'intérieur ou sur les bords de l'ellipse), le plus proche de $X\hat{\beta}$ (il s'agit de l'estimateur ridge) n'a aucune de ses deux coordonnées nulles, à la fois pour la contrainte 1 et pour la contrainte 0.25. Par contre, dans le cas de la régression lasso (fig. 8.2b), nous voyons que parmi les points qui satisfont la contrainte (*i.e.* à l'intérieur ou sur les bords du losange), le plus proche de $X\hat{\beta}$ (il s'agit de l'estimateur lasso) a pour coordonnées dans le repère X_1, X_2 $(0, 1)'$ ou $(0, 0.5)'$ selon la contrainte envisagée. Dans ces deux cas,

la variable X_1 a pour coefficient 0, elle n'est pas sélectionnée. Seule une contrainte plus grande que 1 permettrait à β_1 d'être non nul. Ainsi, le choix de $J(.)$ permet de sélectionner ou non des variables donnant ainsi des modèles d'interprétations différentes.
Les fonctions de régularisation les plus utilisées pénalisent les vecteurs β qui ont de trop fortes coordonnées. Elles sont basées sur des normes ou des mélanges de normes.

— Régression ridge, pénalité l^2 : $J(\beta) = \|\beta\|^2 = \sum_{j=1}^{p} \beta_j^2$: pénalise beaucoup les vecteurs β qui présentent de fortes coordonnées.
— Régression lasso : $J(\beta) = \|\beta\|_1^2 = \sum_{j=1}^{p} |\beta_j|$: pénalise les vecteurs β qui présentent de fortes coordonnées et conduit aussi à une sélection de variables (voir discussion ci-dessous).
— Régression elastic-net : $J(\beta) = \alpha\|\beta\|_1^2 + (1-\alpha)\|\beta\|^2$ qui permet d'avoir un compromis entre les deux solutions ci-dessus, au prix d'un coefficient supplémentaire à choisir α.
— Régression group lasso : les coefficients sont naturellement regroupés en K groupes distincts et on souhaite retenir tout un groupe ou l'éliminer mais on ne souhaite pas éliminer qu'une variable dans un groupe. La pénalité est alors $J(\beta) = \sum_{k=1}^{K} \lambda_k \|\beta^{(k)}\|_2$ où $\beta^{(k)}$ est sous-vecteur des coefficients correspondant au groupe k.
— Régression fused lasso $J(\beta) = \alpha\|\beta\|_1^2 + (1-\alpha)\sum_{j=1}^{p-1} |\beta_{j+1} - \beta_j|$: permet de sélectionner à la fois un certain nombre de variables mais aussi de limiter les variations entre coefficients de variables consécutives.

8.2 Problème du centrage-réduction des variables

Nous avons toujours considéré jusqu'à présent le modèle général

$$Y = X\beta + \varepsilon$$

et avons supposé que l'une des variables explicatives pouvait être la constante $\mathbb{1}$. Jusqu'à présent cette variable avait le même rôle que les autres variables explicatives potentielles. Le rôle de l'intercept est différent et il est d'usage de l'inclure dans le modèle sauf situation très particulière. Pour cette raison, nous allons désormais considérer le modèle suivant

$$Y = \mu\mathbb{1} + X\beta + \varepsilon. \tag{8.3}$$

La valeur d'un coefficient β_j dépend de l'échelle des mesures de la variable explicative associée X_j : par exemple, β_j sera différent si la variable est mesurée en grammes ou en kilogrammes. Lors du calcul de la norme, afin de ne pas pénaliser ou favoriser un coefficient, il est souhaitable que chaque coefficient soit affecté de manière « semblable ». Une manière classique de réaliser cet équilibre consiste à centrer et réduire toutes les variables de X. Une variable centrée-réduite $\widetilde{X}_j$ issue de la variable X_j s'écrit

$$\widetilde{X}_j = (X_j - \bar{x}_j\mathbb{1})/\hat{\sigma}_{X_j},$$

où $\bar{x}_j$ est la moyenne empirique de X_j (*i.e.* $\sum_{i=1}^n x_{ij}/n$) et $\hat{\sigma}^2_{X_j}$ une estimation de la variance (par exemple $\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2/n$). *La matrice* $\widetilde{X}$ *contiendra donc des variables centrées réduites.* Le modèle (8.3) devient alors

$$Y = \tilde{\mu}\mathbb{1} + \widetilde{X}\widetilde{\beta} + \varepsilon. \tag{8.4}$$

Le coefficient $\tilde{\mu}$ associé à la variable $\mathbb{1}$, appelé coefficient constant (ou *intercept* en anglais), est un coefficient qui joue un rôle particulier. Il permet aux prévisions du modèle envisagé de se situer autour de la moyenne de Y, de localiser le problème. Puisqu'elle sert à localiser le problème, cette variable n'est donc pratiquement *jamais inclus* dans la contrainte de norme.
Les variables $\tilde{X}$ sont centrées (et réduites), elles sont donc toutes orthogonales à la variable $\mathbb{1}$. Comme la variable $\mathbb{1}$ est exclue de la contrainte et qu'elle est orthogonale aux autres, son coefficient estimé par une régression sous contrainte type lasso ou ridge est simplement la moyenne empirique des observations de Y : $\widehat{\tilde{\mu}} = \bar{y}$ (voir l'exercice 8.6 p. 212).
Après avoir estimé les paramètres avec $\mathbb{1}$ et $\widetilde{X}$ et Y, il est possible de prévoir une nouvelle valeur $x'_{n+1} = (x_{n+1,1}, \cdots, x_{n+1,p})$ grâce à la formule suivante :

$$\hat{y}^p_{n+1} = \widehat{\tilde{\mu}} + \sum_{j=1}^p \left(\frac{x_{n+1,j} - \bar{x}_j}{\hat{\sigma}_{X_j}} \right) \widehat{\tilde{\beta}}_j. \tag{8.5}$$

Remarquons que cette prévision peut s'écrire comme une combinaison linéaire des variables initiales

$$\hat{y}^p_{n+1} = \left(\widehat{\tilde{\mu}} - \sum_{j=1}^p \bar{x}_j \frac{\hat{\beta}_j}{\hat{\sigma}_{X_j}} \right) + \sum_{j=1}^p x_{n+1,j} \frac{\hat{\beta}_j}{\hat{\sigma}_{X_j}}. \tag{8.6}$$

A partir de maintenant et afin d'avoir les notations les plus simples possibles, nous considérerons que les variables de la matrice X sont centrées et réduites et que le modèle s'écrit (on enlève les « tildes ») :

$$Y = \mu\mathbb{1} + X\beta + \varepsilon. \tag{8.7}$$

8.3 Ridge, lasso et elastic-net

La plupart des pénalités utilisées pour régulariser le critère des moindres carrés sont basées sur les normes 1 et 2. Ce sont des variantes des estimateurs ridge et lasso. Nous présentons dans cette partie les principales différences et similitudes entre ces deux approches. Pour un $\lambda > 0$ donné, considérons les problèmes de minimisation suivants :

$$(\hat{\mu}, \hat{\beta}_{\text{ridge}}(\lambda)) = \underset{\mu \in \mathbb{R}, \beta \in \mathbb{R}^p}{\text{argmin}} \left\{ \|Y - \mu\mathbb{1} - X\beta\|^2 + \lambda\|\beta\|^2 \right\}$$

et

$$(\hat{\mu}, \hat{\beta}_{\text{lasso}}(\lambda)) = \underset{\mu \in \mathbb{R}, \beta \in \mathbb{R}^p}{\text{argmin}} \left\{ \|Y - \mu \mathbb{1} - X\beta\|^2 + \lambda \|\beta\|_1 \right\}.$$

Les variables X_j étant centrées (réduites), elles sont orthogonales au vecteur $\mathbb{1}$, l'estimateur de μ ne dépend pas de λ et vaut toujours $\bar{y}$ (voir exercice 8.6).
Rappelons les propriétés principales de l'estimateur des MCO (non pénalisé) et de l'estimateur ridge. Ce dernier a été introduit en section 4.1, p. 73 avec une formule explicite :

$$\hat{\beta}_{\text{ridge}}(\lambda) = (X'X + \lambda I)^{-1} X'Y. \tag{8.8}$$

i) *Existence et unicité* : l'ajustement (via les MCO) $\hat{Y} = P_X Y + \bar{y}\mathbb{1}$ existe et est unique et l'on peut toujours trouver un (et surtout plusieurs) $\hat{\beta}$ qui satisfont $\hat{Y} = X\hat{\beta} + \bar{y}\mathbb{1}$. Si l'hypothèse $\mathcal{H}_1$ est vérifiée, alors le vecteur $\hat{\beta}$ est unique et nous avons la formule explicite $\hat{\beta} = (X'X)^{-1}X'Y$.
Pour la régression ridge, grâce à la formule (8.8), nous avons, dès que $\lambda > 0$ fixé, existence et unicité de l'estimateur ridge (il s'agit du résultat d'une fonction de λ), sans avoir recours à l'hypothèse $\mathcal{H}_1$. Cependant $\hat{Y}_{\text{ridge}}(\lambda) = X\hat{\beta}_{\text{ridge}}(\lambda) + \bar{y}\mathbb{1}$ n'est pas une projection (voir exercice 8.2).

ii) *Nullité de toutes les coordonnées du vecteur des coefficients* : pour les MCO, sauf cas particulier, il n'y a aucune raison que toutes les coordonnées soient nulles.

Il en est de même pour l'estimateur ridge, dès que $\lambda > 0$ fini, nous avons $\hat{\beta}_{\text{ridge}}(\lambda) \neq 0$ (sauf dans le cas très particulier ou $X'Y = 0$).

iii) *Biais* : l'estimateur des MCO est sans biais et l'estimateur ridge est biaisé.

iv) *Variance* : nous avons déjà calculé la variance de l'estimateur des MCO et de celui de l'estimateur ridge (pour ce dernier voir équation (4.6) p. 77). Nous avons montré que $\mathrm{V}(\hat{\beta}_{MCO}) \geq \mathrm{V}(\hat{\beta}_{ridge})$.

Dans le cas de la régression lasso, λ étant considéré comme fixé, l'estimateur (ou les estimateurs s'il n'est pas unique) est l'argument du minimum de la fonction $h(\beta) = \|Y - \mu\mathbb{1} - X\beta\|^2 + \lambda\|\beta\|_1$. Intéressons-nous au point i) *Existence et unicité*. Cette fonction est la somme de deux fonctions convexes (puisque $\lambda > 0$ est fixé) et elle est donc convexe. Nous en déduisons qu'elle a donc un minimum. De plus, l'ensemble des points qui réalisent ce minimum noté $\{\hat{\beta}_{\text{lasso}}(\lambda)\}$ est non vide, assurant l'existence.
Pour les questions d'unicité, nous renvoyons le lecteur vers l'exercice 8.7, mais les conclusions sont identiques aux MCO : $X\hat{\beta}_{\text{lasso}}(\lambda)$ est unique mais il faut ajouter l'hypothèse $\mathcal{H}_1'$: X_ξ est de plein rang, pour garantir l'unicité du vecteur $\hat{\beta}_{\text{lasso}}(\lambda)$ (avec X_ξ est la matrice X limitée aux colonnes $j \in \{1, \ldots, p\}$ pour lesquelles $|z_j| = \lambda$, avec $z \in \partial q$ vérifiant l'équation (8.9)).
Ensuite il serait pratique d'avoir une formule donnant directement $\hat{\beta}_{\text{lasso}}$, comme celle (8.8) pour $\hat{\beta}_{\text{ridge}}$ (ou celle de $\hat{\beta}$), ce qui nous permettrait de faire les calculs et de déduire les propriétés statistiques iii) et iv) et de conclure sur ii). Rappelons que

pour obtenir cette dernière équation, il suffit de partir de la fonction à minimiser, de la dériver et d'annuler sa dérivée. Pour le lasso, la fonction h n'est pas différentiable pour tout β (prendre exemple en $0_p \in \mathbb{R}^p$). Ce problème peut être contourné en prenant le sous-différentiel mais l'équation obtenue ne permet pas de trouver $\hat{\beta}_{\text{lasso}}$ avec une formule explicite (sauf dans le cas orthogonal) et un algorithme itératif doit être mis en œuvre.

Le plus populaire, introduit par Fu (1998), est un algorithme de descente coordonnée par coordonnée. Comme nous allons le voir, si nous fixons toutes les coordonnées de β sauf la j^{e}, nous pouvons trouver facilement une écriture explicite de cette coordonnée $[\hat{\beta}_{\text{lasso}}(\lambda)]_j$ en fonction des données et des autres coordonnées. L'idée de cet algorithme est alors d'effectuer cette minimisation tour à tour sur chacune des coordonnées j et de s'arrêter quand l'algorithme ne progresse plus. Montrons donc que, si nous supposons que toutes les coordonnées de β sauf la j^{e} sont fixées, alors nous avons une formule explicite pour celle-ci. Remarquons que $\beta_j \mapsto h(\beta)$ est convexe et en plus dérivable dès que $\beta_j \neq 0$. Nous avons alors, en laissant de côté la notation $\hat{\beta}_{\text{lasso}}(\lambda)$ pour une plus légère $\hat{\beta}$, que la dérivée (par rapport à β_j) est nulle en $\hat{\beta}_j$:

$$-2X_j'(Y - \bar{y}\mathbb{1} - \sum_{k \neq j} \hat{\beta}_k X_k) + 2\hat{\beta}_j X_j' X_j + \lambda \frac{\hat{\beta}_j}{|\hat{\beta}_j|} = 0 \quad \text{si} \quad \hat{\beta}_j \neq 0.$$

En notant $R_j = X_j'(Y - \bar{y}\mathbb{1} - \sum_{k \neq j} \hat{\beta}_k X_k)$, nous obtenons

$$2R_j = 2\hat{\beta}_j X_j' X_j + \lambda \frac{\hat{\beta}_j}{|\hat{\beta}_j|} \quad \text{si} \quad \hat{\beta}_j \neq 0.$$

Puisque $\lambda > 0$ et $X_j' X_j > 0$, nous en déduisons que le signe de R_j est le même que celui de $\hat{\beta}_j$ et nous pouvons donc remplacer $\frac{\hat{\beta}_j}{|\hat{\beta}_j|}$ par $\frac{R_j}{|R_j|}$, ce qui nous donne

$$\hat{\beta}_j = \frac{R_j}{X_j' X_j}(1 - \lambda \frac{1}{2|R_j|}) \quad \text{si} \quad \hat{\beta}_j \neq 0.$$

Puisque $\hat{\beta}_j$ est du même signe que R_j, afin de garantir cette condition dans l'équation précédente, nous devons considérer uniquement la partie positive du facteur le plus à droite :

$$\hat{\beta}_j = \frac{R_j}{X_j' X_j}(1 - \lambda \frac{1}{2|R_j|})_+$$

Nous obtenons donc l'algorithme 1.

Algorithme 1 Régression Lasso par descente coordonnées par coordonnées (avec les variables X centrées réduites).

1. **Initialisation** : fixer $\beta^0 \in \mathbb{R}^p$, $k = 0$.
2. **Répéter**
   - Pour $j = 1, \ldots, p$ faire :
     $\beta_j^{k+1} \leftarrow \frac{R_j}{X_j' X_j}(1 - \frac{\lambda}{2|R_j|})_+$ avec $R_j = X_j'(Y - \bar{y}\mathbb{1} - \sum_{k \neq j} \beta_k^k X_k)$
   - $k \leftarrow k + 1$

jusqu'à $\beta^{k+1} \approx \beta^k$.

Une autre possibilité consiste à utiliser une version modifiée de l'algorithme Lars (voir section 8.6, p. 213).
Rappelons que l'estimateur lasso est l'argument du minimum de la fonction. Pour une fonction convexe $x \mapsto f(x)$, on a alors l'équivalence suivante :

$$\hat{x} \in \operatorname*{argmin}_{x \in \mathbb{R}^p} f(x) \Leftrightarrow 0 \in \partial f(\hat{x})$$

qui, appliquée à notre cas, nous indique qu'en un des estimateurs lasso $\hat{\beta}_{\text{lasso}}(\lambda)$, le sous-différentiel de h, noté $\partial h(\hat{\beta}_{\text{lasso}}(\lambda))$, contient le vecteur $0 \in \mathbb{R}^p$.
En utilisant le fait que le sous-différentiel d'une somme de deux fonctions convexes est la somme des deux sous-différentiels et que le sous-différentiel d'une fonction différentiable (*i.e.* la fonction $\beta \mapsto h_1(\beta) = \|Y - \mu\mathbb{1} - X\beta\|^2$) est le singleton limité au gradient (donc ici $\partial h_1(\beta) = \{\nabla h_1(\beta)\} = \{-2X'(Y - \mu\mathbb{1} - X\beta)\}$), nous avons alors que notre condition s'écrit

$$0 \in \{-2X'(Y - \mu\mathbb{1} - X\hat{\beta}_{\text{lasso}}(\lambda))\} + \partial q(\hat{\beta}_{\text{lasso}}(\lambda))$$

où $q(\beta) = \lambda\|\beta\|_1$. Enfin, en utilisant le fait que le sous-différentiel de cette norme est calculable comme

$$z \in \partial q(\beta) \Leftrightarrow \begin{cases} z_j = \lambda \operatorname{sign}(\beta_j) = \lambda \frac{\beta_j}{|\beta_j|} & \text{si } \beta_j \neq 0, \\ z_j \in [-\lambda, \lambda] & \text{si } \beta_j = 0, \end{cases}$$

nous avons donc qu'une condition nécessaire et suffisante pour que $\hat{\beta}_{\text{lasso}}(\lambda)$ soit un argument du minimum de h s'écrit

$$-2X'(Y - \mu\mathbb{1} - X\hat{\beta}_{\text{lasso}}(\lambda)) + z = 0 \tag{8.9}$$

avec $z \in \partial q(\beta)$. Malheureusement, et comme annoncé, cette équation ne donne pas une écriture explicite de $\hat{\beta}_{\text{lasso}}(\lambda)$. Cependant, nous en déduisons tout de même un fait très intéressant sur le lasso. De l'équation (8.9), nous avons, avec $z \in \partial q(\beta)$:

$$X'X\hat{\beta}_{\text{lasso}}(\lambda) = 2X'Y - z,$$

ce qui en prémultipliant par $\hat{\beta}'_{\text{lasso}}(\lambda)$ donne

$$0 \leq \hat{\beta}'_{\text{lasso}}(\lambda)X'X\hat{\beta}_{\text{lasso}}(\lambda) = \hat{\beta}'_{\text{lasso}}(\lambda)(2X'Y - z).$$

Si nous appelons ξ l'ensemble des variables explicatives pour lesquelles le coefficient de $\hat{\beta}'_{\text{lasso}}(\lambda)$ n'est pas nul, nous avons alors en remplaçant z par sa valeur :

$$0 \leq \sum_{j\in\xi} [\hat{\beta}_{\text{lasso}}(\lambda)]_j (2[X'Y]_j - \lambda \operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j)).$$

Pour que cette condition nécessaire soit vérifiée, il faut donc absolument que
— si $[\hat{\beta}_{\text{lasso}}(\lambda)]_j$ est positif alors $2[X'Y]_j > \lambda \operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j) \geq 0$;
— si $[\hat{\beta}_{\text{lasso}}(\lambda)]_j$ est négatif alors $2[X'Y]_j < \lambda \operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j) \leq 0$.
En prenant le plus grand élément en valeur absolue du vecteur $X'Y$ (noté $\|X'Y\|_\infty = \max_j |[X'Y]_j|$), nous avons que si $\lambda \geq 2\|X'Y\|_\infty$ alors aucun des deux points ci-dessus ne peut être vérifié, signifiant que pour $\lambda \geq 2\|X'Y\|_\infty$ nous avons $\hat{\beta}_{\text{lasso}}(\lambda) = 0$.
En conclusion, l'homologue du point ii) pour le lasso est le suivant : *si $\lambda \geq 2\|X'Y\|_\infty$ alors le vecteur $\hat{\beta}_{\text{lasso}}(\lambda)$ a toutes ses coordonnées nulles, aucune variable n'est sélectionnée.*
Dès que la valeur de λ passe sous ce seuil, la première variable, celle dont l'indice correspond à $\|X'Y\|_\infty$, est ajoutée au modèle. Rappelons que si les variables de X et Y sont centrées réduites, $X'Y$ représente à $1/n$ près la corrélation entre chaque variable de X et la variable Y. Cela correspond dans ce cas à la variable explicative la plus corrélée avec Y, c'est-à-dire la même variable ajoutée que dans une sélection ascendante partant d'un modèle avec juste la constante.
Enfin pour les propriétés statistiques iii) et iv), comme nous n'avons pas de formule explicite pour l'estimateur lasso (sauf cas orthogonal), il est plus difficile de les obtenir. Nous renvoyons à Giraud (2014) pour plus de précisions.

8.3.1 Régressions avec la package glmnet

On rappelle que, pour $\alpha \in [0,1]$ et $\lambda \geq 0$ fixés, les estimateurs elastic net sont définis par

$$(\hat{\mu}, \hat{\beta}(\lambda)) = \underset{\mu\in\mathbb{R},\beta\in\mathbb{R}^p}{\operatorname{argmin}} \left\{\|Y - \mu\mathbb{1} - X\beta\|^2 + \lambda J(\beta)\right\}$$

avec

$$J(\beta) = \alpha\|\beta\|_1^2 + (1-\alpha)\|\beta\|^2.$$

Le paramètre α régule le compromis entre les pénalités lasso et ridge : pour $\alpha = 1$ on obtient les estimateurs lasso tandis qu'on aura les estimateurs ridge pour $\alpha = 0$.
Le package **glmnet** permet de calculer et de visualiser les estimateurs obtenus. Nous le présentons à travers l'exemple des données `ozone` :

```
> ozone <- read.table("ozone.txt",header=TRUE,sep=";",row.names=1)
> ozone <- ozone[, -c(11:12)]
```

Il n'est pas possible d'utiliser de formule dans la fonction **glmnet** pour spécifier la variable à expliquer et les variables explicatives. Il faut renseigner les variables explicatives dans une matrice. On utilise souvent la fonction **model.matrix** pour obtenir cette matrice :

```
> ozone.X <- model.matrix(O3 ~ ., data = ozone)[,-1]
> ozone.Y <- ozone$O3
```

Si, parmi les variables explicatives, il y a des variables qualitatives, la fonction **model.matrix** fait un codage disjonctif des variables qualitatives. Ce codage remplace donc une variable admettant I modalités en I variables binaires. Pour chaque variable, la somme des variables issues du codage donne le vecteur constant dont toutes les valeurs valent 1. La fonction **model.matrix** supprime la colonne associée au codage de la première modalité afin qu'il n'y ait pas de problème de colinéarité.

Par défaut, la fonction **glmnet** choisit une grille de valeurs possibles pour λ et calcule les estimateurs pour chaque valeur dans la grille. Le choix de la pénalité s'effectue à travers l'argument `alpha` de **glmnet**. On pourra par exemple obtenir les estimateurs ridge, lasso et elastic net (avec $\alpha = 0.5$) avec les ordres suivants :

```
> ridge <- glmnet(ozone.X, ozone.Y, alpha = 0)
> lasso <- glmnet(ozone.X, ozone.Y)#par défaut alpha=1
> en <- glmnet(ozone.X, ozone.Y, alpha = 0.5)
```

Il est souvent d'usage de représenter les valeurs des estimateurs en fonction de λ ou de la norme $\|\hat{\beta}(\lambda)\|_1$. Un tel graphe est appelé *chemin de régularisation*. On peut par exemple visualiser les chemins de régularisation des estimateurs précédents à l'aide des commandes suivantes (voir figure 8.3) :

```
> plot(ridge,main="Ridge",ylim=c(-2,2))
> plot(ridge,xvar="lambda",main="Ridge",ylim=c(-2,2))
> plot(lasso,main="Lasso",ylim=c(-2,2))
> plot(lasso,xvar="lambda",main="Lasso",ylim=c(-2,2))
> plot(en,main="Elastic net",ylim=c(-2,2))
> plot(en,xvar="lambda",main="Elastic net",ylim=c(-2,2))
```

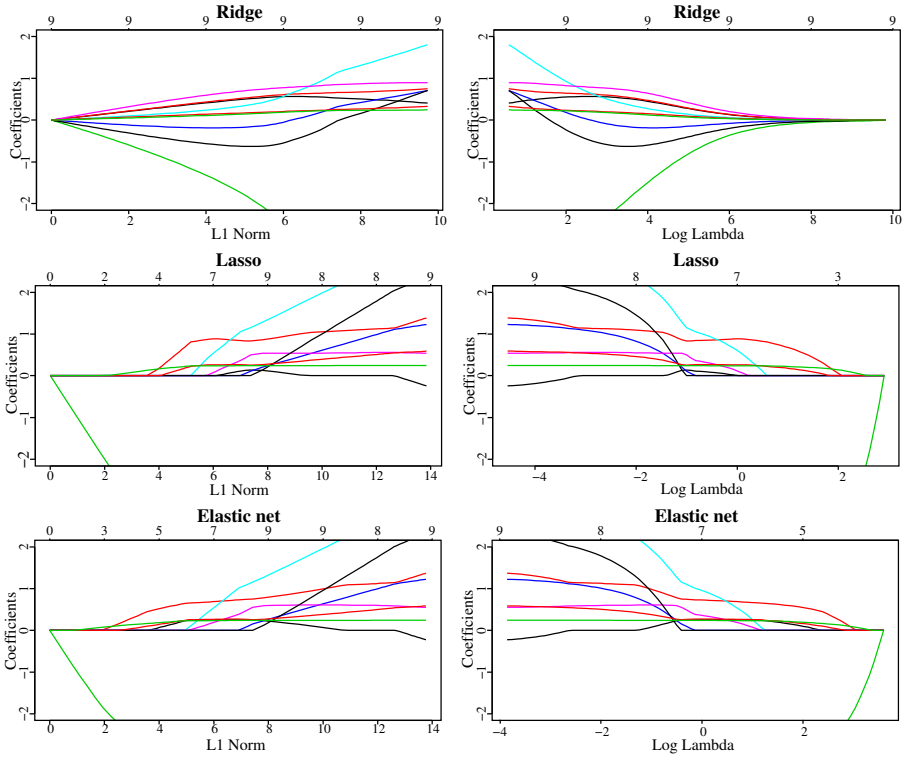


Fig. 8.3 – Chemins de régularisation ridge (gauche), lasso (milieu) et elastic net (bas) en fonction de $\|\hat{\beta}(\lambda)\|_1$ (gauche) et $\log(\lambda)$ (droite).

Les graphes de gauche représentent les valeurs des coefficients en fonction de la norme 1 du vecteur de coefficients tandis que ceux de droite montrent ces mêmes coefficients en fonction du logarithme de λ. Il y a bien entendu une « symétrie » entre ces deux représentations : une faible norme pour les paramètres correspond à une forte valeur de λ. On remarque que la présence de la norme 1 dans la pénalité (pour le lasso et elastic net) a tendance à mettre à 0 un certain nombre d'estimateurs. En effet, tous les coefficients sont à 0 lorsque λ est grand, ils « quittent 0 » les uns après les autres au fur et à mesure que λ diminue. Ces approches permettent donc de faire de la sélection de variables puisque pour une valeur de λ fixée, un certain nombre d'estimateurs seront égaux à 0. Les valeurs affichées en haut des graphes de la figure 8.3 indiquent le nombre de coefficients non nuls.

Remarque

Les valeurs de coefficients représentées sur la figure 8.3 sont les valeurs calculées dans l'échelle des variables initiales. Ces valeurs se déduisent des valeurs obtenues sur les données centrées-réduites à partir de l'équation (8.6).

8.3.2 Interprétation géométrique

Nous repartons de la formulation « avec contrainte ». Dans le cas où $\delta = 0$, tous les coefficients sont à 0 puisqu'il s'agit du seul point à satisfaire la contrainte de norme nulle. Au fur et à mesure que la contrainte sur la norme diminue (on augmente alors δ), les coefficients augmentent et lorsque la valeur de δ est suffisamment grande, les coefficients deviennent les coefficients obtenus par moindres carrés.
Reprenons l'exemple du tableau 8.1. Nous avons déjà présenté une représentation graphique du problème dans $\mathbb{R}^n$ (voir les figures 8.1 et 8.2), prolongeant les représentations du cadre de la régression multiple au cas avec contrainte.
Une autre représentation géométrique est possible. Dans les cas ridge et lasso, nous cherchons la valeur de β dans une boule de rayon δ qui minimise le critère des moindres carrés, cette valeur ayant été nommée $\hat{\beta}_{\text{ridge}}$ ou $\hat{\beta}_{\text{lasso}}$ selon la contrainte choisie. Les ensembles de niveau de la fonction $\beta \mapsto \|Y - X\beta\|^2$ étant des ellipses centrées en $\hat{\beta}$ (estimateur des MCO non contraint), on cherche donc l'élément de la boule (pour la norme 1 ou 2) de rayon δ qui intersecte l'ensemble de plus faible niveau de cette fonction.
Nous avons la représentation de la fonction $\beta \mapsto \|Y - X\beta\|^2$ avec une contrainte de norme de 0.5 donnée en figure 8.4

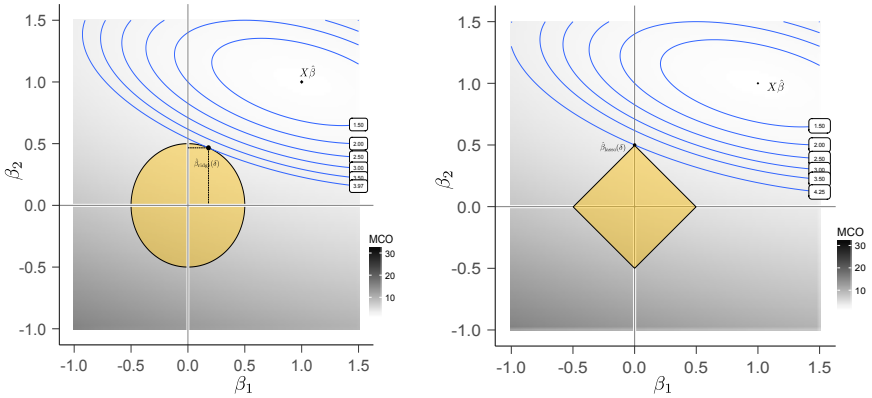


Fig. 8.4 – Représentation des estimateurs ridge (gauche) et lasso (droite) sur un exemple à deux variables. $\hat{\beta}$ correspond à l'estimateur des MCO, les ellipses sont des exemples de lignes de niveau de la fonction $\beta \mapsto \|Y - X\beta\|^2$ et les ensembles des boules pour la norme 2 (gauche) et 1 (droite). Les estimateurs ridge et lasso correspondent à l'intersection de la plus grande ellipse avec les boules.

Pour une contrainte de norme de $\delta = 0.5$, la régression ridge donne un vecteur $\hat{\beta}_{\text{ridge}}(\delta)$ dont aucune des deux coordonnées n'est nulle, alors que la régression lasso donne le vecteur $\hat{\beta}_{\text{lasso}}(\delta) = (0, 0.5)'$. La première variable n'est donc pas retenue, il y a une sélection des variables. Quand la norme δ va augmenter (et dépasser 1), la première variable sera retenue.
Cette remarque se généralise à un nombre p de variables (mentalement car la

représentation devient impossible) : tous les coefficients sont à 0 lorsque $\delta = 0$, puis les coefficients « quittent 0 » les uns après les autres lorsque δ augmente. Dans le cas de la régression ridge, les coefficients quittent simultanément 0 lorsque δ quitte 0. On pourra consulter Giraud (2014) pour plus de détails.

8.3.3 Simplification quand les X sont orthogonaux

Lorsque la matrice X est orthogonale (donc $X'X = I_p$), les estimateurs des MCO et ridge ont une écriture simplifiée :

$$\begin{aligned}\hat{\beta} =& (X'X)^{-1}X'Y = X'Y \\ \hat{\beta}_{\text{ridge}}(\lambda) =& (X'X + \lambda I)^{-1}X'Y = \frac{X'Y}{1+\lambda}.\end{aligned}$$

L'estimateur ridge est une version contractée de l'estimateur des MCO : la j^{e} composante de l'estimateur ridge vaut $\hat{\beta}_j/(1+\lambda)$ où $\hat{\beta}_j$ est la j^{e} composante de l'estimateur des MCO et donc chacune de ses coordonnées a été divisée par $1 + \lambda > 1$ (dès que $\lambda > 0$). La régression ridge revient donc à « diminuer » les estimateurs MCO, à l'image de l'estimateur de James-Stein (8.3.3 p. 205).

On désigne par $\hat{\beta}_{\text{lasso}}(\lambda)$ l'estimateur lasso obtenu par

$$\hat{\beta}_{\text{lasso}}(\lambda) = \underset{\beta}{\operatorname{argmin}} \|Y - X\beta\|^2 + \lambda\|\beta\|_1.$$

Nous nous sommes placés ici dans le cas où X est une matrice orthogonale, cette hypothèse permet d'obtenir une formule explicite pour l'estimateur lasso. Ainsi, l'équation (8.9) s'écrit

$$2\hat{\beta}_{\text{lasso}}(\lambda) = 2X'Y - z$$

ce qui devient pour chaque coordonnée j

$$[\hat{\beta}_{\text{lasso}}(\lambda)]_j = [X'Y]_j - \frac{z_j}{2}.$$

Si la coordonnée $[\hat{\beta}_{\text{lasso}}(\lambda)]_j$ n'est pas nulle, nous avons que $z_j = \lambda \operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j)$, ce qui nous donne pour chaque coordonnée non nulle :

$$[\hat{\beta}_{\text{lasso}}(\lambda)]_j = [X'Y]_j - \frac{\lambda \operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j)}{2} = [X'Y]_j - \frac{\lambda \operatorname{sign}([X'Y]_j)}{2}.$$

Si la coordonnée est positive, nous en déduisons que $[X'Y]_j > 0$ et de plus que $[X'Y]_j > \lambda/2$. Si la coordonnée est négative, nous en déduisons que $[X'Y]_j < 0$ et que $[X'Y]_j < -\lambda/2$. Nous avons donc que $[X'Y]_j$ est de même signe que $[\hat{\beta}_{\text{lasso}}(\lambda)]_j$, ce qui nous permet de remplacer $\operatorname{sign}([\hat{\beta}_{\text{lasso}}(\lambda)]_j)$ par le signe de $[X'Y]_j$, valant $\frac{[X'Y]_j}{|[X'Y]_j|}$:

$$[\hat{\beta}_{\text{lasso}}(\lambda)]_j = [X'Y]_j(1 - \frac{\lambda}{2|[X'Y]_j|})_+$$

et nous gardons la partie positive du facteur le plus à droite afin de garantir que les signes soient bien identiques.
Rappelons que l'estimateur des MCO dans ce cas vaut $\hat{\beta}_j = [X'Y]_j$. Ainsi la j^{e} composante de l'estimateur lasso vaut

$$\text{sign}(\hat{\beta}_j)(|\hat{\beta}_j| - \lambda/2)_+$$

où $(x)_+ = \max(x, 0)$. Le lasso met à 0 les composantes pour lesquelles l'estimateur MCO est en valeur absolue plus petit que $\lambda/2$. Les autres composantes sont simplement les composantes de l'estimateur MCO correspondantes rétrécies vers 0 de $\lambda/2$.
La figure 8.5 représente le comportement des estimateurs ridge et lasso en fonction de la valeur de l'estimateur MCO. On parle de soft tresholding pour ridge et de hard tresholding pour lasso.

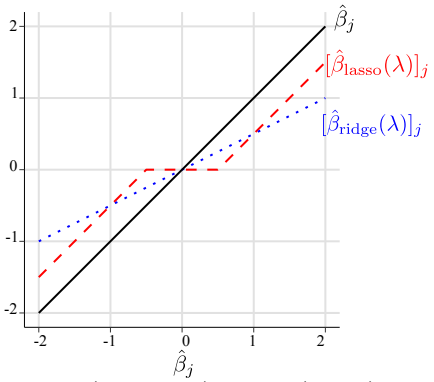


Fig. 8.5 – Estimateurs ridge (pointillés), lasso (tirets) et MCO (trait plein) en fonction de l'estimateur MCO avec $\lambda = 1$.

Analysons l'effet d'un rétrécissement et considérons les estimateurs

$$\tilde{\beta}_i = \frac{1}{1+\lambda}\hat{\beta}_i,$$

où λ est une constante positive à déterminer. Nous avons les propriétés suivantes :

$$\begin{aligned}
\mathbb{E}\tilde{\beta}_i &= \frac{1}{1+\lambda}\beta_i \\
\text{V}(\tilde{\beta}_i) &= \frac{1}{(1+\lambda)^2}\sigma^2 \\
\text{EQM}(\tilde{\beta}_i) &= \frac{1}{(1+\lambda)^2}(\lambda^2\beta_i^2 + \sigma^2).
\end{aligned}$$

James et Stein ont proposé l'estimateur de James-Stein défini par (Lehmann & Casella, 1998, pp. 359 et 368)

$$\hat{\beta}_{JS,i} = \left(1 - \frac{(p-2)\sigma^2}{\|\hat{\beta}\|^2}\right)\hat{\beta}_i.$$

Ils ont démontré que la trace de l'EQM de l'estimateur $\hat{\beta}_{JS}$ était plus petite que la trace de l'EQM de l'estimateur des MC $\hat{\beta}$ lorsque p est plus grand que 2.
Enfin, si l'on prend uniquement la partie positive du premier terme, on obtient un estimateur de James-Stein tronqué

$$\hat{\beta}_{JST,i} = \max\left(0, \left[1 - \frac{(p-2)\sigma^2}{\|\hat{\beta}\|^2}\right]\hat{\beta}_i\right),$$

et l'estimateur est encore amélioré en termes d'EQM. Cet estimateur combine le rétrécissement et le seuillage. En effet lorsque $(p-2)\sigma^2/\|\hat{\beta}\|^2$ est plus grand que 1, le coefficient associé vaut alors 0.
Remarquons que, selon la définition de ces deux estimateurs, ils reviennent tous deux à « rétrécir » les coordonnées de $\hat{\beta}$ vers 0 d'une même grandeur et donc à contraindre la norme de $\hat{\beta}$. En suivant cette idée, il est intéressant d'envisager de contraindre la norme de l'estimation afin d'obtenir des estimateurs possédant un meilleur pouvoir prédictif. Nous avons vu que l'estimateur de James-Stein (tronqué ou non) est un de ces estimateurs. Nous allons détailler d'autres types de contraintes classiques : l'estimateur des moindres carrés sous contrainte de norme, tels que la régression ridge (Hoerl & Kennard, 1970), ou le lasso (Tibshirani, 1996).
Tout d'abord, si l'on souhaite contraindre la norme du coefficient à estimer, il est naturel de supposer que cette norme est inférieure à un nombre δ fixé. Le problème de régression s'écrit alors comme la recherche de $\tilde{\beta}$ tel que

$$\tilde{\beta} = \underset{\beta\in\mathbb{R}^p, \|\beta\|^2\leq\delta}{\operatorname{argmin}} \|Y - X\beta\|^2.$$

Cette méthode revient à la régression ridge (Hastie *et al.*, 2001) dont le principe est exposé à la section 4.1 (p. 73). Géométriquement, cela revient à chercher dans une boule de contrainte de rayon δ, le coefficient $\tilde{\beta}$ le plus proche au sens des moindres carrés.
Les méthodes de régression PLS et de régression sur composantes principales, projetant sur un sous-espace de $\Im(X)$ reviennent aussi à contraindre la norme de $\hat{Y}$. Il est aussi possible de montrer que la méthode PLS revient à contraindre la norme de $\hat{\beta}$ vers 0 (De Jong, 1995). Ces deux méthodes sont exposées au chapitre 9.
A l'image de la régression ridge, il est possible de contraindre non plus la norme euclidienne (au carré) $\|\beta\|^2$, mais la norme de type l^1, à savoir $\|\beta\|_1 = \sum_{i=1}^p |\beta_i|$. Si l'on utilise cette contrainte, la méthode, appelée Lasso, revient à trouver le minimum $\tilde{\beta}$ défini par

$$\tilde{\beta} = \underset{\beta\in\mathbb{R}^p, \|\beta\|_1\leq\delta}{\operatorname{argmin}} \|Y - X\beta\|^2.$$

Notons enfin que ces méthodes permettent à la fois d'obtenir une prévision fiable (moins variable) et de sélectionner des variables. Classiquement elles sont particulièrement indiquées lorsque les variables explicatives sont corrélées. Cependant, nous avons vu que le MSE de la prévision est diminué par l'estimateur de James-Stein, et ce dans tous les cas, lorsque l'hypothèse de normalité est vérifiée. Il semble

donc assez cohérent de penser que les estimateurs contraignant la norme du coefficient à estimer β donneront de meilleures prévisions que l'estimateur des moindres carrés, et ce dans de nombreux cas de figure.

8.3.4 Choix du paramètre de régularisation λ

Le choix du paramètre λ (ou δ) est crucial pour la performance de la procédure. Il va réguler le compromis biais-variance des estimateurs. Lorsque λ est grand, on augmente le poids de la pénalité dans le critère à optimiser. On va donc obtenir des estimateurs plus contraints avec moins de variance mais un biais qui risque d'être élevé (et réciproquement lorsque λ est petit). Il convient donc de trouver des procédures qui permettent de choisir ce paramètre. Les méthodes classiques consistent à fixer une grille de valeurs possibles de λ et à choisir la valeur de λ dans la grille qui minimise une erreur de prévision.
La fonction **cv.glmnet** du package **glmnet** propose d'effectuer ce choix par validation croisée par bloc (cf. algorithme 3, p. 238). Le jeu de données est découpé en K blocs. Pour chaque valeur λ de la grille :

1. on estime les paramètres $\hat{\beta}(\lambda)$ en utilisant $K-1$ blocs ;
2. on prévoit les valeurs sur le bloc non utilisé.

Ces deux étapes sont répétées pour chaque bloc, donnant une prévision $\hat{y}_i(\lambda)$ pour chaque observation et pour chaque λ de la grille. L'erreur de prévision s'obtient en confrontant ces prévisions aux valeurs observées. Deux fonctions sont généralement utilisées (et implémentées dans **glmnet**) :
— l'erreur quadratique de prévision $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i(\lambda))^2$;
— l'erreur absolue de prévision $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i(\lambda)|$.
On sélectionnera la valeur de λ qui minimise l'erreur choisie. Nous utilisons cette fonction pour choisir λ dans les modèles ridge, lasso et elastic net présentés dans la section 8.3.1.

```
> cv.ridge <- cv.glmnet(ozone.X, ozone.Y, alpha = 0)
> cv.lasso <- cv.glmnet(ozone.X, ozone.Y) #alpha=1 par défaut
> cv.en <- cv.glmnet(ozone.X, ozone.Y, alpha = 0.5)
```

La fonction retourne, pour chaque valeur de λ testée :
— l'erreur quadratique de prévision (`cvm`) ainsi qu'une estimation de son écart-type (`cvsd`). On peut en déduire un intervalle de confiance (`cvlo` et `cvup`) associé à cette erreur ;
— le nombre de coefficients non nuls (`nzero`).
La valeur de λ qui minimise l'erreur (`lambda.min`) est également bien entendu proposée. La fonction renvoie de plus une autre valeur `lambda.1se` qui correspond à la plus grande valeur de λ pour laquelle l'erreur se situe à plus un écart type de l'erreur en `lambda.min`. En pratique, cela signifie que l'utilisateur peut choisir `lambda.min` ou `lambda.1se`. Si on privilégie la parcimonie du modèle (lorsqu'on

fait du lasso par exemple), on choisira `lambda.1se`. On obtient ces deux valeurs de λ avec :

```
> cv.ridge$lambda.min
[1] 4.209389
> cv.ridge$lambda.1se
[1] 24.65448
```

On peut visualiser les erreurs en fonction de $\log(\lambda)$ avec les commandes suivantes (voir figure 8.6) :

```
> plot(cv.ridge, main = "Ridge")
> plot(cv.lasso, main = "Lasso")
> plot(cv.en, main = "Elastic net")
```

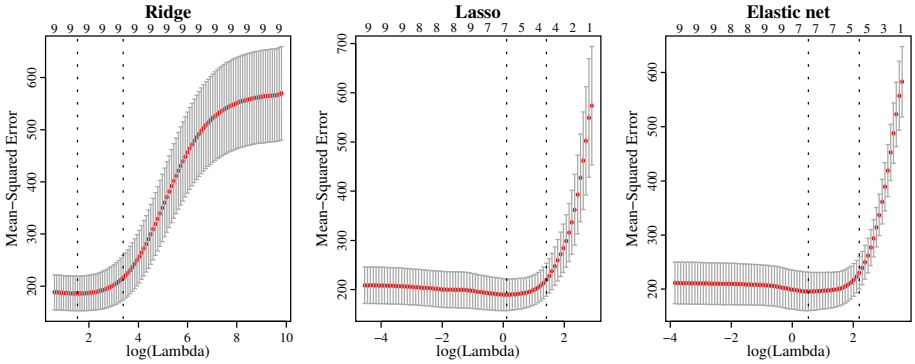


Fig. 8.6 – Erreurs quadratiques de prévision en fonction de $\log(\lambda)$ pour les estimateurs ridge (gauche), lasso (milieu) et elastic net (droite).

On remarque que deux lignes verticales sont représentées sur le graphe. Celle de gauche correspond à la valeur `lambda.min`, celle de droite à `lambda.1se`. Une fois le paramètre λ choisi, l'estimateur final est ensuite ajusté avec toutes les données. On rappelle que les données ont été centrées-réduites pour calculer les estimateurs $\hat{\beta}(\lambda)$. Ainsi la prévision d'un nouvel individu $x'_{n+1} = (x_{n+1,1}, \ldots, x_{n+1,p})$, s'effectuera selon

$$\hat{y}^p_{n+1}(\lambda) \quad = \quad \bar{y} + \sum_{j=1}^{p} \left(\frac{x_{n+1,j} - \bar{x}_j}{\hat{\sigma}_{X_j}} \hat{\beta}_j(\lambda) \right).$$

On remarque que cette prévision peut se réécrire comme une combinaison des variables initiales :

$$\hat{y}^p_{n+1}(\lambda) = \left(\bar{y} - \sum_{j=1}^{p} \bar{x}_j \frac{\hat{\beta}_j(\lambda)}{\hat{\sigma}_{X_j}} \right) + \sum_{j=1}^{p} x_{n+1,j} \frac{\hat{\beta}_j(\lambda)}{\hat{\sigma}_{X_j}}.$$

Sur R, il suffira d'appliquer la fonction **predict** à l'objet construit avec **cv.glmnet**. On obtiendra la prévision pour les deux nouveaux individus

```
> xnew
      T12  T15 Ne12 N12 S12 E12 W12   Vx   O3v
[1,] 13.6 14.4    1   0   0   1   0 3.55  97.8
[2,] 21.8 23.6    6   4   0   0   0 2.50 112.0
```

avec

```
> predict(cv.ridge,newx=xnew)
            1
[1,] 90.89298
[2,] 90.87307
```

Par défaut, c'est la valeur `lambda.1se` qui est utilisée pour faire la prévision. On pourra utiliser la valeur `lambda.min` en utilisant l'option `s` dans la fonction **predict**.

8.4 Intégration de variables qualitatives

Il arrive fréquemment que, parmi les variables explicatives, certaines soient qualitatives. Ce problème a largement été abordé dans le chapitre 6 où nous avons vu que la solution classique consiste à utiliser un codage disjonctif complet : chaque modalité est transformée en un vecteur d'indicatrices d'appartenance à la modalité. Nous avons par exemple vu sur l'exemple des eucaplytus que la variable `bloc`, qui prenait 3 modalités, était recodée de la façon suivante

$$\texttt{bloc} = A = \begin{bmatrix} A1 \\ A1 \\ A2 \\ A2 \\ A3 \\ A3 \end{bmatrix} \quad \Longrightarrow \quad A_c = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Un tel codage revient à utiliser un coefficient pour chaque modalité de la variable dans le modèle et conduit à une surparamétrisation du modèle qui n'est alors plus identifiable. Les solutions proposées dans le chapitre 6 consistaient à utiliser des contraintes identifiantes comme fixer à 0

— le coefficient associé à une modalité, appelée *modalité de référence* ;
— la somme des coefficients associés à la variable qualitative.

Pour les estimateurs MCO, nous avons montré que le choix de ces contraintes a une influence sur la valeur des estimateurs, et donc l'interprétation des coefficients mais pas sur le modèle. Les prévisions sur des nouveaux individus seront par exemple toujours les mêmes, quelle que soit la contrainte. En est-il de même avec les régressions pénalisées ?

Nous répondons à cette question à l'aide de l'exemple suivant. Nous considérons à nouveau les données `ozone` qui comportent deux variables qualitatives parmi les variables explicatives (`nebulosite` à 2 modalités et `vent` à 4 modalités) et nous proposons deux paramétrisations différentes. La fonction **model.matrix** utilise par défaut la première modalité dans l'ordre alphabétique comme modalité de référence : `NUAGE` pour la variable `nebulosite` et `EST` pour la variable `vent`.

```
> ozone<-read.table("ozone.txt",header=TRUE,sep=";",row.names = 1)
> library(glmnet)
> ozone.X <- model.matrix(O3~.,data=ozone)
> ozone.Y <- ozone$O3
> cv.defaut <- cv.glmnet(ozone.X,ozone.Y)
> lassodefaut<-glmnet(ozone.X,ozone.Y,lambda=cv.defaut$lambda.min)
```

On utilise maintenant `NORD` comme modalité de référence pour la variable `vent`

```
> ozone$vent <- relevel(ozone$vent,ref="NORD")
> ozone.X <- model.matrix(O3~.,data=ozone)
> cv.nord <- cv.glmnet(ozone.X,ozone.Y)
> lassonord <- glmnet(ozone.X,ozone.Y,lambda=cv.nord$lambda.min)
```

Les coefficients (`lassodefaut$beta` et `lassonord$beta`) ainsi que les λ sélectionnés par validation croisée sont différents. Comme nous pouvons le voir sur les quatre premières observations les prévisions sont également différentes :

```
> predict(lassodefaut,ozone.X[1:4,])
               s0
19960422 78.60837
19960429 91.08168
19960506 74.16593
19960514 75.37471
> predict(lassonord,ozone.X[1:4,])
               s0
19960422 79.39104
19960429 90.66975
19960506 74.42717
19960514 75.26528
```

Nous pouvons donc conclure que, contrairement aux MCO, le choix de la contrainte identifiante a une influence sur les prévisions pour les méthodes pénalisées. Même si le plus souvent les résultats obtenus seront très proches, on peut bien entendu se poser la question du choix de la contrainte. Il n'y a pas de réponse évidente à cette question.

La contrainte somme peut paraître la plus naturelle puisqu'elle ne repose pas sur une modalité particulière :

```
> ozone.X <- model.matrix(O3~.-vent-nebulosite+C(vent,sum)+
                              C(nebulosite,sum),data=ozone)
> cv.sum <- cv.glmnet(ozone.X,ozone.Y)
> lassosum <- glmnet(ozone.X,ozone.Y,lambda=cv.sum$lambda.min)
> predict(lassosum,ozone.X[1:4,])
              s0
19960422 78.13964
19960429 90.22089
19960506 74.88035
19960514 74.66571
```

Pour les variables qualitatives, il est souvent recommandé d'utiliser le Group-lasso (voir section 13.3). Dans ce cas on regroupe les coefficients associés à chaque variable qualitative. Le Group-lasso aura ainsi tendance à mettre à 0 tous les coefficients associés à la variable.

8.5 Exercices

Exercice 8.1 (Questions de cours)

1) La régression avec contrainte de norme sur β est en général utilisée lorsque l'hypothèse ci-dessous n'est pas satisfaite :
   A. $\mathcal{H}_1$ concernant le rang de X (matrice du plan d'expérience),
   B. $\mathcal{H}_2$ concernant l'espérance et la variance des résidus,
   C. $\mathcal{H}_3$ concernant la normalité des résidus.
2) Lorsque la matrice $(X'X)$ n'est pas inversible, l'estimateur des moindres carrés
   A. existe et est unique,
   B. existe et n'est pas unique,
   C. n'existe pas, aucun estimateur ne minimise les moindres carrés.
3) La régression ridge peut être vue comme une régression avec comme critère d'estimation les moindres carrés et une contrainte de norme sur
   A. le plan d'expérience (X),
   B. les paramètres,
   C. aucun rapport.
4) La régression lasso peut être vue comme une régression avec comme critère d'estimation les moindres carrés et une contrainte de norme sur
   A. le plan d'expérience (X),
   B. les paramètres,
   C. aucun rapport.
5) Parmi les affirmations suivantes, lesquelles sont vraies ?
   A. Les méthodes régularisées de type lasso/ridge permettent de réduire la variance des estimateurs MCO,
   B. On utilise généralement les méthodes ridge/lasso lorsque le nombre de variables explicatives p est grand,
   C. Les méthodes régularisées de type lasso/ridge permettent de réduire le biais des estimateurs MCO,
   D. Les estimateurs ridge/lasso sont toujours plus performants que les estimateurs MCO.

6) Soit $\lambda \geq 0$. Les estimateurs lasso s'obtiennent en pénalisant le critère des moindres carrés par

A. $\lambda \sum_{j=1}^{p} |\beta_j|$,
B. $\lambda \sum_{j=1}^{p} \sqrt{\beta_j}$,
C. $\lambda \sum_{j=1}^{p} \beta_j^2$,
D. $\lambda \sum_{j=1}^{p} \log(\beta_j^2)$,
E. $\lambda \sum_{j=1}^{p} \log(|\beta_j|)$,
F. $\lambda \sum_{j=1}^{p} \beta_j$.

7) On considère les estimateurs lasso définis par la pénalité proposée à la question précédente. Parmi les affirmations suivantes, lesquelles sont vraies ?
A. Les estimateurs seront proches de 0 pour de très petites valeurs de λ,
B. Les estimateurs seront proches des estimateurs MCO pour de très grandes de λ,
C. Les estimateurs seront proches de 0 pour de très grandes valeurs de λ,
D. Les estimateurs seront proches des estimateurs MCO pour de très petites de λ,
E. Il faut toujours choisir λ le plus grand possible,
F. Il faut toujours choisir λ le plus petit possible.

Exercice 8.2 (Projection et régression ridge)
Soit le modèle de régression classique

$$Y = X\beta + \varepsilon.$$

Montrer que l'estimateur ridge n'est pas une projection.

Exercice 8.3 (Variance des valeurs ajustées avec une régression ridge)
Soit le modèle de régression classique

$$Y = X\beta + \varepsilon.$$

Considérons les valeurs ajustées $\hat{Y}$ obtenues avec l'estimateur des MCO et les valeurs ajustées $\hat{Y}_{ridge}(\lambda)$ obtenues avec l'estimateur ridge (pour un $\lambda \geq 0$ donné).
Montrer que

$$\|\hat{Y}_{ridge}\| \leq \|\hat{Y}\|.$$

Exercice 8.4 (Nombre effectif de paramètres de la régression ridge)
Toutes les variables sont centrées et réduites. Dans la régression multiple sur p variables explicatives, le nombre de coefficients inconnus $\{\beta_j\}$ est p, c'est-à-dire $\mathrm{tr}(P_X)$. Rappelons que l'application qui à Y fait correspondre $\hat{Y}$ est P_X. La trace de cette application donne le nombre effectif de paramètres. Cette notion peut être étendue à la régression ridge.
1) Dans le cas de la régression ridge, quelle est l'application $H(\kappa)$ qui à Y fait correspondre $\hat{Y}_{\mathrm{ridge}}(\kappa)$?
2) Soit A une matrice carrée symétrique $p \times p$ (donc diagonalisable). Montrer que si U_j est vecteur propre de A associé à la valeur propre d_j^2, alors U_j est aussi vecteur propre de $A + \lambda I_p$ associé à la valeur propre $\lambda + d_j^2$.
3) En utilisant la décomposition en valeurs singulières de X : $X = QDP'$ avec Q et P matrice orthogonale et $D = \mathrm{diag}(d_1, \ldots, d_p)$, montrer que

$$\mathrm{tr}(X(X'X + \lambda I_p)^{-1}X') = \mathrm{tr}(PD(D^2 + \lambda I_p)^{-1}DP').$$

En déduire que le nombre effectif de paramètres de la régression ridge est

$$\sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda}.$$

Exercice 8.5 (Estimateurs à rétrécissement – *shrinkage*)
Soit le modèle de régression classique

$$Y = X\beta + \varepsilon.$$

Soit la décomposition en valeurs singulières de X :

$$PXQ' = D = \begin{pmatrix} \Delta \\ 0 \end{pmatrix},$$

où P et Q sont 2 matrices orthogonales de dimension $n \times n$ et $p \times p$ et Δ est la matrice diagonale des valeurs singulières $\{\delta_i\}$ de dimension p. Posons $Z = PY$, $\gamma = Q\beta$ et $\eta = P\varepsilon$.

1) Etablir que si $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_n)$, alors

$$Z = D\gamma + \eta,$$

$Z_{1:p} \sim \mathcal{N}(\Delta\gamma, \sigma^2 I_p)$ et $Z_{(p+1):n} \sim \mathcal{N}(0, \sigma^2 I_{n-p})$. Ici $Z_{1:p}$ est le vecteur constitué des p premières coordonnées de Z alors que $Z_{(p+1):n}$ contient les $n-p$ dernières.

2) Etablir que la trace de la matrice de l'EQM pour un estimateur linéaire $\hat{\beta} = AY$ de β est la même que celle de $\hat{\gamma} = Q\hat{\beta}$, estimateur de γ.

3) Etablir que l'estimateur des moindres carrés de γ est

$$\hat{\gamma}_{\mathrm{MC}} = \Delta^{-1} Z_{1:p}.$$

et en déduire que $\hat{\gamma}_{\mathrm{MC}} \sim \mathcal{N}(\gamma, \sigma^2 \Delta^{-2})$. L'estimateur $\hat{\gamma}_{\mathrm{MC}}$ est linéaire en Y et ses coordonnées sont indépendantes entre elles.

4) Montrer que l'EQM de la i^{e} coordonnée de $\hat{\gamma}_{\mathrm{MC}}$ vaut σ^2/δ_i^2.

5) Prendre un estimateur linéaire de γ :

$$\hat{\gamma}(c) = \mathrm{diag}(c_i) Z_{1:p}.$$

Vérifier que ses coordonnées sont normales et indépendantes entre elles. Montrer ensuite l'égalité suivante :

$$EQM(\hat{\gamma}(c)_i) = \mathbb{E}(\hat{\gamma}(c)_i - \gamma_i)^2 = c_i^2 \sigma^2 + \gamma_i^2 (c_i \delta_i - 1)^2.$$

6) En déduire que si $\gamma_i^2 < \frac{\sigma^2}{\delta_i^2} \frac{(1/\delta_i) + c_i}{(1/\delta_i) - c_i}$, alors $\mathrm{EQM}(\hat{\gamma}(c)_i) < \hat{\gamma}_{\mathrm{MC}}$.

Pour une condition particulière dépendant des X, il existe un estimateur linéaire de coordonnées indépendantes qui possède un meilleur EQM que celui des MC.

7) Montrer que si $c_i = \frac{\delta_i}{\delta_i^2 + \kappa}$, alors $\hat{\gamma}(c) = Q(X'X + \kappa I_p)^{-1} Q' D' Z$, et en déduire que

$$\hat{\beta} = Q'\gamma = (X'X + \kappa I_p)^{-1} X'Y.$$

Pour une valeur particulière du vecteur c, nous retrouvons l'estimateur ridge. Ce type d'estimateur permet une généralisation de la régression ridge.

Exercice 8.6 (coefficient constant et régression sous contrainte)
Soit le modèle de régression multivarié classique où la p^{e} variable est le vecteur $\mathbb{1}_n$. La régression sous contrainte (avec la contrainte $J()$) est utilisée pour estimer β, avec λ fixé. Le p^{e} coefficient n'est pas inclus dans la contrainte, seules les autres variables $\xi = \{1, \ldots, p-1\}$ de X sont incluses dans la contrainte. Par ailleurs, ces variables de l'ensemble ξ sont toutes centrées.

1) Montrer que $\mathrm{vect}(\mathbb{1}) \overset{\perp}{\oplus} \mathrm{vect}(X_\xi)$.
2) Montrer que $\mathrm{argmin}_\beta \|Y - X\beta\|^2 + \lambda J(\beta_\xi) = \mathrm{argmin}_\beta \|P_X Y - X\beta\|^2 + \lambda J(\beta_\xi)$.
3) Déduire des questions précédentes que $\hat{\beta}_p(\lambda)$ estimé par cette méthode vaut $\bar{y}$.

Exercice 8.7 (Unicité pour la régression lasso, Giraud (2014))
On considère la régression lasso pour $\lambda > 0$ fixé et nous allégeons la notation et notons $\hat{\beta}$ au lieu de $\hat{\beta}_{\text{lasso}}(\lambda)$.
1) En utilisant le fait que $Z \in \mathbb{R}^n, Z \mapsto \|Y - Z\|^2$ est une fonction strictement convexe ($\forall \alpha \in [0,1],\ g(\alpha Y + (1-\alpha)Z) < \alpha g(Y) + (1-\alpha) g(Z)$), montrer que si $\hat{\beta}_1 \in \mathbb{R}^p$ et $\hat{\beta}_2 \in \mathbb{R}^p$ sont solutions de la régression lasso alors nécessairement $X\hat{\beta}_1 = X\hat{\beta}_2$. (Prendre le milieu $(\hat{\beta}_1 + \hat{\beta}_2)2$) et montrer que l'on aboutit à une contradiction si $X\hat{\beta}_1 \neq X\hat{\beta}_2$.)
2) En déduire que $X\hat{\beta}$ est unique.
3) Soit $\hat{\beta}_1 \in \mathbb{R}^p$ et $\hat{\beta}_2 \in \mathbb{R}^p$ deux solutions de la régression lasso avec $(X\hat{\beta}_1 = X\hat{\beta}_2)$. Montrer que forcément les équations (8.9) (p. 198) vérifiées par $\hat{\beta}_1$ et $\hat{\beta}_2$ entraînent que $z_1 \in \partial q$ et $z_2 \in \partial q$ sont égaux (on notera par la suite z ce vecteur).
4) Déduire du point précédent que quelle que soit $\hat{\beta}$ solution de la régression lasso, nous n'avons qu'un seul ensemble $\xi \subset \{1, \ldots, p\}$ pour lequel $\forall j \in \xi,\ |z_j| = 1$.
5) Montrer que cet ensemble ξ contient toutes les coordonnées non nulles de $\hat{\beta}$ solution quelconque de la régression lasso.
6) Soit une solution $\hat{\beta}$ de la régression lasso, montrer que pour les coordonnées dans ξ nous avons

$$X'_\xi X_\xi \hat{\beta}_\xi = [X'Y]_\xi - \frac{\lambda}{2} z_\xi$$

et donc que sous $\mathcal{H}'_1$ (X_ξ est de plein rang) nous avons unicité.

Exercice 8.8
Le fichier `echan_lasso.csv` contient un n échantillon $(X_i, Y_i), i = 1, \ldots, 60$ issu d'un modèle de régression

$$Y = m(X) + \varepsilon.$$

Le fichier `courbe_lasso.csv` contient les valeurs de la fonction à estimer $m : \mathbb{R} \to \mathbb{R}$.
1) Représenter la fonction à estimer ainsi que le nuage de points.
2) Ecrire un modèle linéaire permettant d'estimer la fonction m dans une base de Fourier.
3) Ecrire une fonction **R** qui renvoie une matrice contenant les valeurs $\cos(2k\pi x)$ et $\sin(2k\pi x), k = 0, \ldots, K$ pour un vecteur x et un entier K donnés.
4) Calculer les estimateurs des moindres carrés du modèle écrit à la question 3 (on prendra $K = 25$). Tracer l'estimateur de m. Interpréter.
5) Faire de même en estimant les paramètres à l'aide de la méthode lasso. Interpréter.

8.6 Note : lars et lasso

Cette note permet de faire un lien géométrique entre la méthode du lasso et la sélection de variables ascendante. La méthode appelée *least angle regression* (LARS) permet pratiquement de calculer les valeurs de $\hat{\beta}_{\text{lasso}}(\lambda)$ pour toutes les valeurs de λ avec un coût calculatoire identique à celui de la régression. Rappelons que toutes les variables sont centrées réduites afin de se débarasser du problème de la localisation (le calcul de l'intercept ou coefficient constant) et d'accorder la même importance à chaque variable. Dans le cadre de cet algorithme et afin de nous rapprocher de la présentation de Efron

et al. (2004), nous allons supposer que les variables sont centrées et normées à l'unité, ce qui donne par exemple pour la j^{e} variable explicative :

$$X_j' X_j = 1, \qquad \bar{x}_j = 0.$$

Dans ce cadre-là, la norme équivaut (à $1/n$ près) à la variance, le produit scalaire entre deux variables donne l'angle entre ces deux vecteurs (puisqu'ils ont même norme) et il équivaut (à $1/n$ près) à la corrélation.

Plaçons-nous dans le cas où nous avons seulement deux variables explicatives représentées par la figure 8.7. Si nous utilisons un choix de variables ascendant (forward), nous partons du modèle sans aucune variable. Comme toutes les variables y compris Y sont centrées, nous avons donc que $\hat{Y}^{(0)} = 0$. Ensuite, nous cherchons à ajouter la variable qui donne le plus d'information. Ceci revient à prendre la variable la plus corrélée au résidu du modèle en cours, c'est-à-dire à $Y - \hat{Y}^{(0)}$. Dans l'exemple de la figure 8.7, nous sélectionnons comme première variable la variable X_1 : elle forme l'angle le plus faible avec $\hat{Y}$ donc avec Y. Le modèle ajusté à la première étape est donc

$$\hat{Y}^{(1)} = P_{X_1} Y = X_1 (X_1' X_1)^{-1} X_1' Y = X_1 (X_1' Y) = X_1 \gamma_1 = \gamma_1 X_1,$$

où γ_1 est la corrélation entre Y et X_1 (à $1/n$ près). Nous nous sommes donc déplacés de l'étape précédente $\hat{Y}^{(0)}$ dans la direction de la variable la plus corrélée (X_1) d'une quantité γ_1.

A la seconde étape, nous ajoutons la seconde variable et nous avons

$$\hat{Y}^{(2)} = P_X Y.$$

Nous ajoutons à $\hat{Y}^{(1)}$ le trajet grisé sur la figure 8.7 sur la perpendiculaire à X_1.

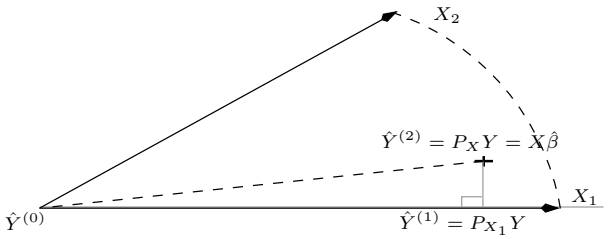


Fig. 8.7 – Sélection ascendante pour deux variables explicatives.

Si nous utilisons la même procédure : sélection de la variable la plus corrélée au résidu courant de l'étape k ($Y - \hat{Y}^{(k)}$) et déplacement dans la direction de cette variable d'une certaine quantité γ, nous avons une autre règle :

$$\hat{Y}^{(k+1)} = \hat{Y}^{(k)} + \gamma X_{j(k)},$$

où $j(k)$ est le numéro de la variable la plus corrélée avec $Y - \hat{Y}^{(k)}$. Si le pas γ est petit (et tend vers 0) nous avons alors le trajet en noir sur la figure 8.8.

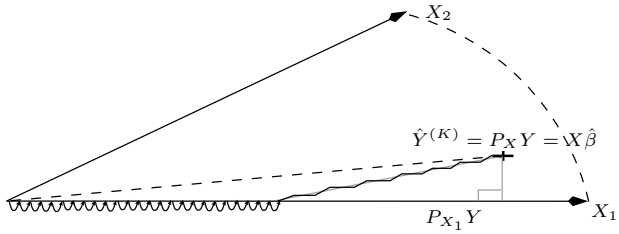


Fig. 8.8 – Procédure « stagewise ».

Numériquement cette procédure n'est pas optimale, car dans les premières étapes nous nous dirigeons selon X_1, puis dans les étapes suivantes nous nous déplaçons autour d'une droite. Cette droite est en fait parallèle à la bissectrice de l'angle entre X_1 et X_2. En effet, pour alterner la direction X_1 puis la direction X_2 comme c'est le cas, il faut être à la bissectrice. Cette bissectrice est la droite telle que son angle (corrélation) avec X_1 est égal à son angle avec X_2.
La procédure lars va permettre d'optimiser ces calculs en 2 étapes. La première étape est le déplacement selon X_1 jusqu'au point $\hat{Y}^{(1)}$, point qui est l'intersection de la parallèle à la bissectrice de X_1 X_2 passant par $\hat{Y}$ (voir fig. 8.9). La seconde est le déplacement sur cette bissectrice jusqu'au point final.
Remarquons enfin que si la variable X_2 était remplacée par son opposé sur le graphique 8.9, donnant ainsi naissance à un nouvel exemple, le chemin vers $\hat{Y}$ resterait le même. Il faudrait calculer l'angle entre $-X_2$ et X_1 pour obtenir la bissectrice.

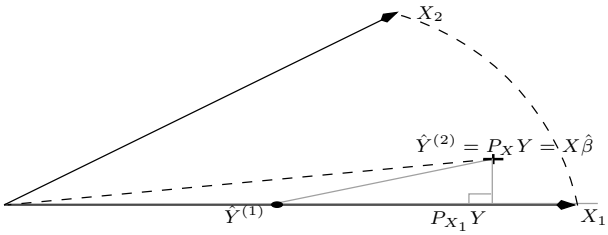


Fig. 8.9 – Procédure « lars ».

Analytiquement, nous avons donc utilisé l'algorithme suivant
1) Calcul du résidu courant : $Y - \hat{Y}^{(k)}$.
2) Détermination de l'ensemble des variables actives $\xi(k)$. Ce sont les variables les plus corrélées avec ce résidu :

$$\xi^{(k)} = \left\{ j \in \{1, \ldots, p\}, X_j(Y - \hat{Y}^{(k)}) = C^{(k)} \right\}$$

avec $C^{(k)}$ le maximum de la valeur absolue de la corrélation (à 1/n près) entre le résidu et les variables : $\max_j |X_j(Y - \hat{Y}^{(k)})|$. Remarquons qu'à la première étape il n'y a qu'une variable dans $\xi^{(k)}$.
3) Déplacement selon le vecteur directeur $\Delta_{\xi^{(k)}}$ qui a un angle (une corrélation) identique avec toutes les variables de $\xi^{(k)}$ (ou à leur opposé) :

$$\hat{Y}^{(k+1)} \quad = \quad \hat{Y}^{(k)} + \gamma_k \Delta_{\xi^{(k)}},$$

avec γ_k choisi comme la plus petite valeur telle qu'une nouvelle variable (ou son opposé) rejoigne l'ensemble $\xi^{(k)}$ des variables actives pour le nouveau résidu $Y - \hat{Y}^{(k+1)}$.

Cet algorithme ne donne pas exactement les solutions de la méthode lasso. Nous retrouvons que la première variable à être intégrée au modèle est celle qui est la plus corrélée. Cependant, une contrainte de non changement de signe doit être ajoutée à l'algorithme afin d'obtenir les solutions du problème du lasso. Nous renvoyons le lecteur intéressé à la lecture de l'article de Efron *et al.* (2004).

Exercice 8.9 (Algorithme LARS)
Soit $s_j^{(k)} = \text{signe}\big\{X_j'(Y - \hat{Y}^{(k)})\big\}$ le signe des corrélations entre chaque variable $j \in \{1, \dots, p\}$ et le résidu de l'étape k. Soit $\mathbb{1}_{\xi^{(k)}}$ le vecteur constitué de 1 et de longueur le nombre de variables dans $\xi^{(k)}$. Notons $\xi^{(k)^c}$ l'ensemble complémentaire de $\xi^{(k)}$ et $X_{\xi^{(k)}}$ la matrice $(\dots, s_j X_j, \dots), j \in \xi^{(k)}$.

1) Vérifier que le vecteur $\tilde{\Delta}_{\xi^{(k)}} = X_{\xi^{(k)}}(X'_{\xi^{(k)}} X_{\xi^{(k)}})^{-1}\mathbb{1}_{\xi^{(k)}}$ possède un produit scalaire constant positif avec toutes les variables de $\xi^{(k)}$.

2) Vérifier que $\|\tilde{\Delta}_{\xi^{(k)}}\|^2 = \mathbb{1}'_{\xi^{(k)}}(X'_{\xi^{(k)}} X_{\xi^{(k)}})^{-1}\mathbb{1}_{\xi^{(k)}}$ et que $\Delta_{\xi^{(k)}} = \tilde{\Delta}_{\xi^{(k)}}/\|\tilde{\Delta}_{\xi^{(k)}}\|$ a pour norme unité. Conclusion : le vecteur directeur (de norme unité) $\Delta_{\xi^{(k)}}$ a bien un angle constant positif avec toutes les variables de $\xi^{(k)}$.

3) Soit le nouvel ajustement

$$\hat{Y}^{(k+1)} = \hat{Y}^{(k)} + \gamma_k \Delta_{\xi^{(k)}}.$$

Calculer le résidu (à l'étape $k+1$) et en déduire que

a) pour les variables de $\xi^{(k)}$ la valeur absolue de la corrélation entre ces variables et le résidu est

$$C^{(k)} - \gamma_k \|\Delta_{\xi^{(k)}}\|;$$

b) pour les variables j qui sont dans $\xi^{(k)^c}$, la corrélation entre ces variables et le résidu est

$$X_j'(Y - Y^{(k)}) - \gamma_k X_j' \Delta_{\xi^{(k)}}.$$

4) Posons que l'unique variable sélectionnée à l'étape $k+1$ est la variable X_j ($j \in \xi^{(k)^c}$). Vérifier à l'aide de la question précédente, que la plus petite valeur de $\gamma_k > 0$ ($i.e$ le plus petit déplacement dans la direction $\Delta_{\xi^{(k)}}$) qui permet à une variable de rejoindre $\xi^{(k)}$ (et de former $\xi^{(k+1)}$) est définie par

$$\min_{l \in \xi^{(k)^c}}{}^{+} \left\{ \frac{C^{(k)} - X_l'(Y - Y^{(k)})}{\|\Delta_{\xi^{(k)}}\| - X_l' \Delta_{\xi^{(k)}}}, \frac{C^{(k)} + X_l'(Y - Y^{(k)})}{\|\Delta_{\xi^{(k)}}\| + X_l' \Delta_{\xi^{(k)}}} \right\},$$

où $\min^+(a, b)$ sélectionne la valeur parmi (a, b) qui est positive. Le minimum est bien sûr atteint avec la variable j.

Chapitre 9

Régression sur composantes : PCR et PLS

L'objectif consiste toujours à expliquer (ou prévoir) la variable Y. Dans ce chapitre, nous allons modifier les variables initiales X et les transformer en « composantes » qui peuvent être vues comme des nouvelles variables, des indicateurs ou des index construits comme des combinaisons linéaires des variables explicatives initiales. Ces approches déplacent donc le problème des variables initiales vers des composantes. L'idée sous-jacente est celle d'un changement de base.

Nous allons choisir des composantes orthogonales entre elles. En effet, travailler avec des variables orthogonales facilite grandement l'analyse d'un problème de régression : pour le calcul de l'estimateur et ses propriétés statistiques, pour le choix de variables (chapitre 7) ou pour la régression sous contraintes (chapitre 8). En général, ces méthodes sont utilisées quand les variables explicatives sont corrélées entre elles et/ou en grand nombre (par exemple lorsque $p > n$). Il est envisageable (et nous verrons cela dans le cas de la régression sur composantes principales) de coupler ces méthodes avec les méthodes de régularisation vues au chapitre précédent.

De nombreuses méthodes existent pour obtenir une base orthogonale de l'espace engendré par les colonnes de X (méthode de Gram-Schmidt, décomposition QR, SVD, diagonalisation...). Ces méthodes d'orthogonalisation dépendent uniquement des variables explicatives et sont indépendantes de Y. Nous présenterons d'abord la méthode de régression sur composantes principales basée sur une diagonalisation de matrice symétrique. Nous étudierons ensuite la régression PLS qui utilise Y pour orthogonaliser l'espace engendré par les colonnes de X.

Ces composantes étant bâties comme des combinaisons linéaires des variables explicatives, il est d'usage de centrer-réduire au préalable ces variables. C'est pourquoi, comme pour les méthodes pénalisées, nous travaillerons avec le modèle suivant

$$Y = \mu \mathbb{1} + X\beta + \varepsilon,$$

où les variables X sont maintenant centrées réduites (voir 8.2, p. 194).

9.1 Régression sur composantes principales (PCR)

9.1.1 Changement de base

La matrice $X'X$ (de dimension $p \times p$) est une matrice symétrique à coefficients réels, elle est donc orthogonalement diagonalisable dans $\mathbb{R}$ et nous pouvons écrire

$$X'X = P\Lambda P', \tag{9.1}$$

où P est la matrice des vecteurs propres normalisés de $(X'X)$, c'est-à-dire que P est une matrice orthogonale ($P'P = PP' = I_p$) et $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_p)$ est la matrice diagonale des valeurs propres classées par ordre décroissant, $\lambda_1 \geq \cdots \geq \lambda_p$.

Remarque
Si l'on effectue l'analyse en composantes principales (ACP) du tableau X (ou du triplet $(X, I_p, I_n/n)$), la matrice P est la matrice des axes principaux normés à l'unité, mais les valeurs propres de l'ACP sont λ_j/n avec j variant de 1 à p.

Rappelons que les variables explicatives X sont centrées réduites et que le modèle de régression de départ est

$$Y = \mu\mathbb{1} + X\beta + \varepsilon.$$

Puisque $P'P = PP' = I_p$, nous pouvons remplacer X par XPP' et nous avons alors

$$Y = \mu\mathbb{1} + XPP'\beta + \varepsilon,$$

que nous décidons de réécrire sous la forme simplifiée suivante :

$$Y \quad = \quad \mu\mathbb{1} + X^*\beta^* + \varepsilon, \tag{9.2}$$

où $\beta^* = P'\beta$ et $X^* = XP$. Les colonnes de X^* sont traditionnellement appelées composantes ou composantes principales [1]. Cette dernière équation (9.2) définit un modèle de régression que nous appellerons modèle « étoile » qui est tout simplement la régression sur les composantes principales X^*.
Par construction, nous avons

$$X^{*\prime}X^* = P'X'XP = P'P\Lambda P'P = \Lambda. \tag{9.3}$$

Cela signifie que les nouvelles variables $X_j^* = XP_j$ constituant les colonnes de X^*, sont orthogonales entre elles et de norme λ_j. C'est une propriété classique des composantes principales d'une ACP.

1. Lors de l'ACP du tableau X (ou du triplet $(X, I_p, I_n/n)$), les composantes principales normées à la valeur propre obtenues sont égales aux vecteurs X_j^* que l'on obtient ici, d'où le nom de la méthode.

9.1.2 Estimateurs des MCO

L'estimateur des MCO de $(\mu, \beta^{*\prime})'$ du modèle étoile ci-dessus est donné par la formule classique

$$(\hat{\mu}, \hat{\beta}^{*\prime})' \quad = \quad ((\mathbb{1}|X^*)'(\mathbb{1}|X^*))^{-1}(\mathbb{1}|X^*)'Y. \tag{9.4}$$

Les variables de X sont centrées et donc orthogonales à $\mathbb{1}$, on a alors

$$\mathbb{1}'X = 0 = \mathbb{1}'XP = \mathbb{1}X^*.$$

Cela permet d'écrire le produit suivant comme une matrice bloc :

$$(\mathbb{1}|X^*)'(\mathbb{1}|X^*) = \begin{pmatrix} n & 0 \\ 0 & X^{*\prime}X^* \end{pmatrix}.$$

L'inverse de cette matrice diagonale par bloc est simplement l'inverse de chaque bloc. Le bloc $X^{*\prime}X^*$ à inverser est déjà calculé en (9.3), son inverse est simplement la matrice diagonale des inverses Λ^{-1} ; les estimateurs des MCO s'écrivent :

$$\begin{aligned} \hat{\mu} &= \frac{1}{n}\sum_{i=1}^{n} y_i, \\ \hat{\beta}^* &= (X^{*\prime}X^*)^{-1}X^{*\prime}Y = \Lambda^{-1}P'X'Y. \end{aligned}$$

Remarquons que la dernière ligne de calcul ci-dessus ne comporte qu'une inversion d'une matrice diagonale, ce qui simplifie grandement son calcul. La matrice de variance covariance de $\hat{\beta}^*$ vaut $\sigma^2((\mathbb{1}|X^*)'(\mathbb{1}|X^*))^{-1}$, ce qui en calculant bloc par bloc donne ici

$$\begin{aligned} \mathrm{V}(\hat{\mu}) &= \frac{\sigma^2}{n} \\ \mathrm{V}(\hat{\beta}^*) &= \sigma^2(X^{*\prime}X^*)^{-1} = \Lambda^{-1}\sigma^2, \end{aligned}$$

En termes statistiques, les estimateurs des coefficients $\hat{\beta}^*_j, j = 1, \ldots, p$ associés à chacune des composantes principales sont non corrélés (puisque la matrice Λ est diagonale). De plus, les valeurs propres étant classées par ordre décroissant, la variance des estimateurs est donc dans l'ordre opposé : $\mathrm{V}(\hat{\beta}^*_j) \leq \mathrm{V}(\hat{\beta}^*_k)$ si $j < k$. Ainsi les estimateurs obtenus avec les premières composantes sont moins variables que les estimateurs obtenus sur les dernières composantes.
En nous rappelant que la variance de l'estimateur des MCO est

$$\mathrm{V}(\hat{\beta}) \quad = \quad \sigma^2(X'X)^{-1} = \sigma^2 P\Lambda^{-1}P',$$

dans le modèle de régression nous n'avons donc aucune assurance de non-corrélation entre coefficients, ni aucun ordre de variabilité.
Rappelons enfin que $\hat{\beta}^*$ minimise aussi les moindres carrés puisque les moindres carrés du modèle étoile et du modèle initial sont identiques par construction :

$$\|Y - \mu\mathbb{1} - X\beta\|^2 \quad = \quad \|Y - \mu\mathbb{1} - XPP'\beta\|^2 = \|Y - \mu\mathbb{1} - X^*\beta^*\|^2.$$

Remarque
En général la régression sur composantes principales propose un nombre $k < \text{rang}(X)$ de composantes et donc les moindres carrés obtenus avec la régression linéaire et ceux obtenus avec la régression sur composantes principales sont différents et les prévisions proposées par les deux modèles n'ont aucune raison d'être identiques. Dans ce cas-là, il s'agit de deux modélisations différentes.

9.1.3 Choix de composantes/variables

Une fois ce changement de base effectué, il est important de choisir les composantes qui figureront dans le modèle. Les approches sont similaires aux procédures de choix de variables. Deux grandes familles de méthodes sont généralement utilisées : les méthodes classiques de choix de variables (voir chapitre 7, p. 159) ou celles plus spécifiques utilisant l'ordre de construction des composantes (*i.e.* l'ordre des λ_i décroissant).

Approche choix de variables

Rappelons que des variables orthonormées (et pas simplement orthogonales) simplifient les procédures de sélection de variables (voir l'exercice 7.6 p. 186). La sélection est faite de manière indépendante d'une variable à l'autre : garder ou non une variable n'implique pas de recalculer les coefficients pour les autres variables. Il suffit de calculer les estimations dans le modèle complet puis d'annuler les coefficients des variables non retenues (seuillage dur) en fonction d'un critère choisi. Il n'y a pas besoin d'utiliser un algorithme spécifique (Forward, Backward, Stepwise...) pour trouver le meilleur modèle.
Le cadre des variables orthogonales où se trouve le modèle « étoile » est très voisin. Rappelons que la norme au carré des variables X_j^* vaut λ_j. Pour normer les variables X_j^* il suffit donc de diviser ces variables par $\sqrt{\lambda_j}$, le coefficient associé à la j^{e} variable orthonormée est alors celui de la j^{e} variable orthogonale multiplié par $\sqrt{\lambda_j}$. Il faut donc analyser $|\hat{\beta}_j^* \sqrt{\lambda_j}|$.
Les procédures de choix de composantes « classiques » reviennent alors à dire que la j^{e} variable est conservée si la ligne correspondant à la méthode choisie est vérifiée :

$$\begin{aligned} \text{Test :} \quad & |\hat{\beta}_j^* \sqrt{\lambda_j}| > 2\sigma \\ \text{R}_{\text{a}}^2 : \quad & |\hat{\beta}_j^* \sqrt{\lambda_j}| > \sigma \\ \text{C}_{\text{p}} : \quad & |\hat{\beta}_j^* \sqrt{\lambda_j}| > \sqrt{2}\sigma. \end{aligned}$$

Quelle que soit la procédure choisie, il faut donc une estimation de σ pour sélectionner les coefficients et donc les composantes. Quand l'hypothèse $\mathcal{H}_1$ est vérifiée sur le modèle complet, un estimateur de σ^2 peut être obtenu en utilisant les résidus du modèle complet.
Le calcul de l'AIC ou du BIC nécessite de connaître le nombre de composantes que nous allons sélectionner. La constante ne fait pas partie des composantes à sélectionner. Dans notre cas, la constante est dans le modèle et μ est estimé par

$\hat{\mu}$. Nous proposons donc d'ordonner les composantes en respectant l'ordre des $|\hat{\beta}_j^* \sqrt{\lambda_j}|$. En respectant cet ordre, on aura par exemple pour le BIC

$$\begin{aligned}\mathrm{BIC}(1) &= n(1+\log 2\pi) + n \log \frac{\mathrm{SCR}(1)}{n} + 3\log n \\ &= n(1+\log 2\pi) + n \log \frac{\sum_i (y_i - \bar{y})^2 - \hat{\beta}_{(1)}^{*2}\lambda_{(1)}}{n} + 3\log n\end{aligned}$$

car nous avons estimé 3 paramètres $(\hat{\mu}, \hat{\beta}_{(1)}, \hat{\sigma})$ puis

$$\mathrm{BIC}(2) = n(1+\log 2\pi) + n \log \frac{\sum_i (y_i - \bar{y})^2 - \hat{\beta}_{(1)}^{*2}\lambda_{(1)} - \hat{\beta}_{(2)}^{*2}\lambda_{(2)}}{n} + 4\log n$$

et ainsi de suite en respectant donc l'ordre obtenu en classant les composantes selon les valeurs de $|\hat{\beta}_j^* \sqrt{\lambda_j}|$. Cette procédure n'est pas implémentée dans les logiciels et nous vous proposons de le faire dans l'exercice 9.2. Vous pouvez également faire une fonction qui, après avoir estimé σ^2, effectue un choix de composantes avec les différents critères (voir exercice 9.3).

Approche utilisant la validation croisée

La validation croisée (voir algorithme 3 p. 238) est souvent utilisée pour choisir le nombre de composantes en régression PCR. Elle est implémentée dans la fonction **pcr** du package **pls**. Le principe est toujours le même, à savoir qu'on divise le jeu de données initial en b parties distinctes approximativement de même taille. Pour une partie donnée, par exemple la i^{e}, on met de côté cette i^{e} partie des données pour effectuer la prédiction après avoir estimé les modèles sur toutes les autres observations appelées souvent données d'apprentissage. Et on répète ce travail sur les b parties. Ainsi à la fin de la procédure, tous les individus ont été prévus une fois et il est donc possible d'évaluer la qualité des prévisions. Il faut juste spécifier le critère de qualité. En général le critère proposé est l'erreur quadratique moyenne de prévision (EQMP) qui est la moyenne des erreurs de prévision au carré

$$\mathrm{EQMP}(j) = \frac{1}{n}\sum_{i=1}^{n} (\hat{Y}_i^p(pcr, j) - Y_i)^2,$$

où $\hat{Y}_i^p(pcr, j)$ désigne la prévision de l'observation i avec la régression sur j composantes principales. Le nombre de composantes choisi $\hat{j}$ est celui qui minimise la moyenne des erreurs de prévision au carré $\mathrm{argmin}_j\, \mathrm{EQMP}(j)$. D'autres critères peuvent être utilisés comme

$$\mathrm{MAE}(j) = \frac{1}{n}\sum_{i=1}^{n} |\hat{Y}_i^p(pcr, j) - Y_i|.$$

La fonction **pcr** teste donc les modèles avec 1 composante, puis 2 composantes... en utilisant l'ordre de construction des composantes (*i.e.* l'ordre des λ_j décroissant).

On évalue la qualité des différents modèles testés (ici le modèle 1 utilise uniquement la première composante X_1^*, puis le modèle 2 les 2 premières composantes X_1^*, X_2^*...) en utilisant le critère choisi. La taille optimale $\hat{\jmath}$ est celle qui conduit à la minimisation du critère. Ce travail est proposé dans l'exercice 9.4.

9.1.4 Retour aux données d'origine

Remarquons que choisir des variables revient soit à éliminer les colonnes dans X^* et les coefficients associés dans $\hat{\beta}^*$, soit à mettre des 0 dans les coordonnées du vecteur $\hat{\beta}^*$ dont les variables n'ont pas été choisies. Etant donné ξ un sous-ensemble de $\{1, \ldots, p\}$, nous noterons X_ξ^* (ou P_ξ) la matrice où seules les colonnes de l'ensemble ξ ont été conservées et $\hat{\beta}_\xi^*$ le vecteur de coefficients $\hat{\beta}^*$ dans lequel seules les coordonnées correspondant à l'ensemble ξ ont été gardées (donc élément de $\mathbb{R}^{|\xi|}$). Par abus de notation, nous noterons $\hat{\beta}_\xi^*$ le vecteur de $\mathbb{R}^p$ égal à $\hat{\beta}^*$ excepté pour les variables non retenues et pour lesquelles le coefficient a été mis à 0.
Rappelons que la formule de passage de β^* à β est donnée par

$$\beta^* = P'\beta \tag{9.5}$$

et donc pour passer de $\hat{\beta}^*$ à $\hat{\beta}$ il suffit de calculer $\hat{\beta} = P\hat{\beta}^*$. En appliquant cette formule à $\hat{\beta}_\xi^*$, une fois choisies les composantes pertinentes, nous pouvons revenir au modèle initial

$$\hat{\beta}_\xi \quad = \quad P_\xi \hat{\beta}_\xi^*$$

et ensuite calculer avec les variables initiales les valeurs ajustées par le modèle à composantes choisies :

$$\hat{Y}_\xi \quad = \quad \bar{y}\mathbb{1} + X\hat{\beta}_\xi.$$

Il faut faire attention car la matrice des variables explicatives a été centrée et réduite avant de commencer à travailler comme nous l'avons annoncé en début de chapitre. Les $\hat{\beta}_\xi$ obtenus correspondent à ces dernières, pour revenir aux données initiales notées X^{ini} , écrivons

$$\begin{aligned} \hat{Y}_\xi &= \bar{y}\mathbb{1} + (X^{\text{ini}} - \bar{X}^{\text{ini}})\frac{\hat{\beta}_\xi}{\hat{\sigma}_{X_j^{\text{ini}}}} \\ &= \hat{\mu}_{\text{fin}} + X^{\text{ini}}\hat{\beta}_{\text{fin}}, \end{aligned} \tag{9.6}$$

où $\hat{\beta}_{\text{fin}}$ correspond aux β pour les variables explicatives initiales et $\hat{\mu}_{\text{fin}}$ vaut $\bar{y}\mathbb{1} - \bar{X}^{\text{ini}}\hat{\beta}_{\text{fin}}$. Nous sommes donc revenus aux données initiales (non centrées/réduites). Cette dernière formulation est intéressante pour

— connaître le rôle des variables initiales et ne pas seulement se contenter du modèle dans les composantes qui sont souvent peu explicables en termes métier ;

— effectuer facilement de la prévision avec des nouvelles valeurs de X. Ainsi si nous obtenons une nouvelle valeur $x'_{n+1} = (x_{n+1,1}, \cdots, x_{n+1,p})$, il faut d'abord la centrer et la réduire avec les valeurs des moyennes et des écarts-types empiriques utilisées pour centrer et réduire les variables du tableau initial puis utiliser $\hat{\beta}_\xi$. Nous pouvons prédire y_{n+1} par :

$$\hat{y}^p_{\xi,n+1} = \hat{\mu}_{\text{fin}} + \sum_{j=1}^{p} x_{n+1,j} \left[\hat{\beta}_{\text{fin}}\right]_j. \tag{9.7}$$

9.1.5 La régression sur composantes en pratique

Nous reprenons les données `ozone` :

```
> ozone <- read.table("ozone.txt",header=TRUE,sep=";",row.names=1)
```

Nous pouvons effectuer une régression avec toutes les variables potentiellement explicatives :

```
> modeleinit <- lm(O3 ~ ., data = ozone[,1:10])
> round(coefficients(modeleinit),2)
(Intercept)       T12       T15      Ne12       N12       S12
      54.73     -0.35      1.50     -4.19      1.28      3.17
        E12       W12        Vx       O3v
       0.53      2.47      0.61      0.25
> BIC(modeleinit)
[1] 431.8923
```

En utilisant le package **leaps** et le critère BIC, nous avions retenu 5 variables.

```
> library(leaps)
> choix <- regsubsets(O3 ~ .,nbest=1,nvmax=10,data=ozone[,1:10])
> resume <- summary(choix)
> indmin <- which.min(resume$bic)
> nomselec <- colnames(resume$which)[resume$which[indmin,]][-1]
> formule <- formula(paste("O3~",paste(nomselec,collapse="+")))
> modeleBIC <- lm(formule,data=ozone[,1:10])
> round(coefficients(modeleBIC),2)
(Intercept)         T15        Ne12          Vx         O3v
      61.83        1.06       -3.99        0.31        0.26
> BIC(modeleBIC)
[1] 415.8866
```

Le modèle choisi avec le critère de choix BIC a un BIC plus petit que le modèle initial puisque c'est le critère que nous minimisons. Travaillons maintenant avec les variables orthogonalisées. La première étape consiste à centrer et réduire les variables potentiellement explicatives.

```
> X <- ozone[,2:10]
> Xbar  <- apply(X, 2, mean)
> stdX <- sqrt(apply(X, 2, var))
> Xcr <- scale(X, center = Xbar, scale = stdX)
```

Il faut ensuite « orthogonaliser » la matrice `Xcr` en utilisant la fonction effectuée à l'exercice (9.7).
En utilisant le critère de BIC, nous trouvons 2 composantes (c'est-à-dire que nous projetons dans un sous-espace de dimension 2). Avec les différentes formules présentées dans le chapitre, nous revenons d'abord à $\hat{\beta}$ puis nous estimons $\hat{\mu}$ et $\hat{\sigma}$. Nous obtenons le modèle suivant :

```
(Intercept)       T12       T15      Ne12       N12       S12
      52.79      0.40      0.47     -2.54     -0.87      0.02
        E12       W12        Vx       O3v
       1.06     -0.85      0.23      0.34
```

Nous calculons le BIC de ce modèle avec 4 paramètres (2 coefficients pour les 2 composantes, 1 coefficient correspondant à l'intercept et 1 coefficient pour l'estimateur de σ), il vaut 413.64. Le modèle obtenu est donc meilleur que le modèle obtenu par choix de variables.
Une autre façon de procéder et qui est implémentée dans la fonction **pcr** du package **pls** est un choix séquentiel du nombre de composantes, choix effectué par validation croisée par bloc. Le principe général de validation croisée est expliqué en détail dans la section 10.1 (page 237, du chapitre 10) et son application pour le choix de composantes est exposé en section 9.1.3 (page 221).
Nous contrôlons la graine du générateur afin d'obtenir toujours la même partition pour toutes les méthodes de ce chapitre.

```
> library(pls)
> set.seed(87)
> cvseg <- cvsegments(nrow(ozone), k = 4, type = "random")
### modeles
> modele.pcr <- pcr(O3 ~ ., ncomp=9, data=ozone[,1:10], scale=T,
+                   validation = "CV", segments = cvseg)
> msepcv.pcr <- MSEP(modele.pcr ,estimate=c("train","CV"))
> msepcv.pcr
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps
train        559.8    188.7    186.5    185.2    179.8    158.8
CV           582.9    260.4    260.6    278.2    271.3    239.3
       6 comps  7 comps  8 comps  9 comps
train    152.0    143.9    142.5    139.7
CV       248.1    242.1    244.0    239.4
```

La régression sur composantes principales est conduite simplement grâce à la

fonction `pcr`. Ici nous pouvons avoir au maximum 9 composantes principales ($\min(n_a, p) = p = 9$). La procédure de validation croisée 4 blocs choisit de retenir 5 composantes car l'erreur de prévision (CV) est la plus petite avec 5 composantes. Nous évaluons donc le modèle final.

```
> npcr <- which.min(msepcv.pcr$val["CV",,])-1
> modele.pcr.fin <- pcr(O3 ~ ., ncomp = npcr, scale = TRUE,
+                                          data = ozone[,1:10])
```

Les valeurs des coefficients sont obtenues dans le modèle centré réduit. Afin de retrouver les coefficients dans l'échelle initiale, il faut les diviser par les écarts-types respectifs, puis calculer l'intercept comme indiqué dans l'équation 9.6. Nous obtenons alors

```
(Intercept)       T12       T15      Ne12       N12      S12
       45.1      0.63      0.67     -2.78     -0.16    -0.19
        E12       W12        Vx       O3v
       0.35     -0.85      0.18      0.34
```

Nous calculons le BIC de ce modèle avec 7 paramètres (5 coefficients pour les 5 composantes, 1 coefficient correspondant à l'intercept et 1 coefficient pour l'estimateur de σ), il vaut 422.65. Le modèle obtenu par le package est donc moins bon au sens du BIC que le modèle trouvé en utilisant l'ordre des variables induit par $|\hat{\beta}_j^* \sqrt{\lambda_j}|$.
La décomposition du tableau X en composantes principales ne tient pas compte de la variable cible Y. Il n'y a pas de raison de croire que les vecteurs propres associés aux plus grandes valeurs propres soient les meilleures combinaisons linéaires des variables explicatives pour expliquer Y. Nous allons maintenant proposer une méthode alternative pour construire des vecteurs orthogonaux qui tiennent compte de leur covariance avec la variable Y à expliquer.

9.2 Régression aux moindres carrés partiels (PLS)

Afin de conserver des notations simples, nous considérerons que les variables X et Y sont centrées réduites. A l'image de la régression sur composantes principales, nous sommes intéressés par de nouvelles variables explicatives $t^{(1)}, t^{(2)}, \ldots t^{(k)}$, combinaisons linéaires des variables de départ $t^{(j)} = X\tilde{c}_j$, qui soient orthogonales entre elles et classées par ordre d'importance. Rappelons que les composantes principales X_j^* obéissent à ces mêmes critères. Cependant, le choix de ces composantes $t^{(j)}$ doit être dicté, non pas par la part de variabilité qu'elles représentent parmi les variables explicatives originales (comme en régression sur composantes principales), mais par leur lien avec la variable à expliquer.
Pour cela une procédure itérative va être utilisée.

Définition 9.1
Quand Y est univarié, la régression PLS est appelée PLS1 et se définit itérativement.

- 1re *étape : le tableau X est noté $X^{(1)}$ et Y noté $Y^{(1)}$. La première composante PLS $t^{(1)} \in \mathbb{R}^n$ est choisie telle que*

$$t^{(1)} = \underset{t=X^{(1)}w, w\in\mathbb{R}^p, \|w\|^2=1}{\operatorname{argmax}} < t, Y^{(1)} > .$$

  Ensuite nous effectuons la régression univariée de $Y^{(1)}$ sur $t^{(1)}$ et obtenons

$$Y^{(1)} = r_1 t^{(1)} + \hat{\varepsilon}_1$$

  où le coefficient r_1 estimé par MC vaut $< t^{(1)}, Y^{(1)} > / < t^{(1)}, t^{(1)} >$ et $\hat{\varepsilon}_1 = P_{t^{(1)}\perp} Y^{(1)}$ sont les résidus de la régression simple sans constante ;
- 2e *étape : soit $Y^{(2)} = P_{t^{(1)}\perp} Y^{(1)} = Y^{(1)} - r_1 t^{(1)} = \hat{\varepsilon}_1$ la partie non encore expliquée de Y. Soit $X^{(2)} = P_{t^{(1)}\perp} X^{(1)}$ la partie de $X^{(1)}$ n'ayant pas encore servi à expliquer. La seconde composante PLS est choisie telle que*

$$t^{(2)} = \underset{t=X^{(2)}w, w\in\mathbb{R}^p, \|w\|^2=1}{\operatorname{argmax}} < t, Y^{(2)} > .$$

  Ensuite nous effectuons la régression univariée de $Y^{(2)}$ sur $t^{(2)}$

$$Y^{(2)} = r_2 t^{(2)} + \hat{\varepsilon}_2$$

  où $r_2 \in \mathbb{R}$ est le coefficient de la régression estimé par MC et $\hat{\varepsilon}_2 = P_{t^{(2)}\perp} Y^{(2)}$;
  ...
- k^e *étape : soit $Y^{(k)} = P_{t^{(k-1)}\perp} Y^{(k-1)} = \hat{\varepsilon}_{k-1}$ la partie non encore expliquée de Y. Soit $X^{(k)} = P_{t^{(k-1)}\perp} X^{(k-1)}$ la partie de $X^{(k-1)}$ n'ayant pas encore servi à expliquer. La k^e composante PLS est choisie telle que*

$$t^{(k)} = \underset{t=X^{(k-1)}w, w\in\mathbb{R}^p, \|w\|^2=1}{\operatorname{argmax}} < t, Y^{(k)} > .$$

  Ensuite nous effectuons la régression univariée de $Y^{(k)}$ sur $t^{(k)}$

$$Y^{(k)} = r_k t^{(k)} + \hat{\varepsilon}_k$$

  où $r_k \in \mathbb{R}$ est le coefficient de la régression estimé par MC et $\hat{\varepsilon}_k = P_{t^{(k)}\perp} Y^{(k)}$.

Remarque
La régression PLS cherche une suite de composantes PLS qui soient, par construction, orthogonales. Puisque $t^{(j)}$ est une combinaison linéaire des colonnes de $X^{(j)}$, qui est par construction dans le complément orthogonal de $\Im(t^{(1)}, \ldots, t^{(j-1)})$, le vecteur $t^{(j)}$ sera orthogonal à $t^{(1)}, \ldots, t^{(j-1)}$. Ces composantes sont choisies comme maximisant la covariance (empirique) entre Y et une composante t quand X et Y sont centrées au préalable.

Théorème 9.1
Nous pouvons donc écrire la méthode PLS comme

$$\begin{aligned} Y &= P_{t^{(1)}}Y^{(1)} + \ldots + P_{t^{(k)}}Y^{(k)} + \hat{\varepsilon}_k \\ Y &= r_1 t^{(1)} + \ldots + r_k t^{(k)} + \hat{\varepsilon}_k, \end{aligned}$$

avec $\hat{\varepsilon}_k = P_{t^{(k)}\perp}Y^{(k)} = P_{\Im(t^{(1)},\ldots,t^{(k)})\perp}Y$.

La preuve découle de la définition en notant que les composantes PLS sont orthogonales entre elles.

9.2.1 Algorithmes PLS

A chaque étape nous cherchons à maximiser une fonction sous contrainte. Après introduction du lagrangien, nous avons à chaque étape j la fonction suivante à maximiser :

$$\mathcal{L}(w,\tau) = Y^{(j)\prime}X^{(j)}w - \frac{1}{2}\tau(\|w\|^2 - 1).$$

Le facteur $-1/2$ ne change pas fondamentalement le résultat, mais il permet une simplification des calculs. Une condition nécessaire d'optimum est alors donnée par l'annulation de ses dérivées partielles au point optimum $(w^{(j)}, \tau_j)$ donnant

$$\begin{aligned} X'Y^{(j)} - \tau_j w^{(j)} &= 0 \\ {w^{(j)}}' w^{(j)} &= 1 \end{aligned}$$

La première équation montre que $w^{(j)}$ est colinéaire au vecteur $X'Y^{(j)}$ et la seconde montre qu'il est normé. Si l'on veut un maximum, il suffit de prendre le vecteur $X^{(j)\prime}Y^{(j)}/\|X^{(j)\prime}Y^{(j)}\|$. Le vecteur de signe opposé donnant le minimum.
Lorsque Y est une matrice $n \times q$, les différents algorithmes de PLS correspondent à différentes méthodes de recherche du premier vecteur singulier de $Y'X$: puissance itérée (algorithme nipals), décomposition en valeurs singulières classique (SVD) ou encore diagonalisation de $Y'XX'Y$. Les estimations peuvent donc être sensiblement différentes.

Remarque
L'algorithme nipals propose de calculer la régression PLS même si l'on possède des valeurs manquantes. Pour cela, dès qu'une valeur manquante est rencontrée, elle est ignorée. Ainsi le calcul devient :

$$[Y'X]_j = \sum_{i=1\ldots n, y_i \text{ ou } X_{ij} \text{ non manquants}} y_i X_{ij}$$

ce qui revient, après le centrage et la réduction, à remplacer les valeurs manquantes *dans les données centrées-réduites* par la valeur 0.

9.2.2 Choix de composantes/variables

Le calcul des composantes se fait de façon séquentielle en tenant compte de Y et il faut maintenant choisir le nombre de composantes. En régression PLS, on se focalise sur le nombre de composantes k et on conserve alors les k premières composantes. En général, on recherche une taille de modèle k, ou ici un nombre de composantes k, qui soit compris entre 1 et une taille maximum K. Cette taille maximum peut être choisie comme $K = \text{rang}(X)$ ou comme la taille au-delà de laquelle il est certain que les composantes ne serviront à rien.

Approche choix de variables

Les modèles étant emboîtés, il serait possible d'utiliser les approches choix de variables et une approche que l'on trouve dans la littérature est celle basée sur le calcul d'un AIC ou BIC

$$\begin{aligned} BIC(k) &= n(1+\log 2\pi) + n\log\frac{\text{SCR}(k)}{n} + (k+1)\log n \\ AIC(k) &= n(1+\log 2\pi) + n\log\frac{\text{SCR}(k)}{n} + 2(k+1) \end{aligned}$$

où $\text{SCR}(k)$ est la somme des carrés résiduels dans le modèle avec k composantes.

Approche utilisant la validation croisée

La validation croisée (voir algorithme 3 p. 238) est en général la procédure la plus utilisée en régression PLS. Le principe est toujours le même, à savoir qu'on divise le jeu de données initial en b parties distinctes approximativement de même taille. Pour une partie donnée, par exemple la i^{e}, on met de côté cette i^{e} partie des données pour effectuer la prédiction après avoir estimé les modèles sur toutes les autres observations appelées souvent données d'apprentissage. Et on répète ce travail sur les b parties. Ainsi à la fin de la procédure, tous les individus ont été prévus une fois. On évalue la qualité des différents modèles testés (ici le modèle 1 utilise uniquement la première composante, puis le modèle 2 les 2 premières...) en définissant un critère. Le critère proposé est l'erreur quadratique moyenne de prévision (EQMP) qui est la moyenne des erreurs de prévision au carré

$$\text{EQMP}(j) = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i^p(pls,j) - Y_i)^2.$$

$Y_i^p(pls,j)$ désigne la prévision de l'observation i avec la régression sur j composantes PLS. Le nombre de composantes choisi $\hat{j}$ est celui qui minimise la moyenne des erreurs de prévision au carré $\text{argmin}_j\,\text{EQMP}(j)$. Là encore d'autres critères que l'EQMP (comme le MAE) pourraient être choisis.

9.2.3 Retour aux données d'origine

Une fois le nombre de composantes sélectionné, il est intéressant de revenir aux données initiales. Le modèle exprimé avec les variables $t^{(j)}$ n'est pas facilement interprétable en termes de variables initiales X. PLS va sélectionner un bon sous-espace mais ne sélectionnera pas de variables. Pour interpréter un modèle PLS, il faut revenir aux données initiales. Pour cela, il faut remplacer les composantes $t^{(j)}$ en fonction de $X^{(j)}w^{(j)}$, ce qui fait intervenir non pas les variables explicatives originales, mais celles de l'étape j. Il faut donc ré-exprimer les composantes PLS en fonction du tableau initial de manière itérative, objet du théorème suivant.

Théorème 9.2
Les composantes PLS peuvent s'exprimer en fonction des variables initiales sous la forme de combinaisons linéaires

$$t^{(j)} = X\tilde{w}^{(j)}, \quad 1 \leq j \leq k,$$

où $\tilde{w}^{(j)}$ est défini par

$$\tilde{w}^{(j)} = X \prod_{l=1}^{j} (I - w^{(l)}(t^{(l)\prime}t^{(l)})^{-1}t^{(l)\prime}X)w^{(j)}.$$

La preuve est à faire à titre d'exercice (voir exercice 9.5).
Nous pouvons réécrire le modèle PLS final à k composantes en fonction des variables explicatives.

Théorème 9.3
La méthode PLS à k composantes s'écrit

$$Y = X\hat{\beta}_{\mathrm{PLS}}(k) + \hat{\varepsilon}_k,$$

où $\hat{\varepsilon}_k$ est le résidu final $P_{t^{(k)\perp}}(Y^{(k)}) = P_{\Im(t^{(1)},\ldots,t^{(k)})^{\perp}}(Y)$ et $\hat{\beta}_{\mathrm{PLS}}(k) = r_1\tilde{w}^{(1)} + \ldots + r_k\tilde{w}^{(k)}$.

Afin de retrouver les valeurs ajustées, nous calculons simplement

$$\hat{Y}_{\mathrm{PLS}}(k) = X\hat{\beta}_{\mathrm{PLS}}(k),$$

et si nous voulons revenir aux valeurs initiales (non centrées et réduites) que nous noterons X^{ini} et Y^{ini}

$$\hat{Y}^{\mathrm{ini}}_{\mathrm{PLS}}(k) = \hat{\sigma}_{Y^{\mathrm{ini}}}\left[X\hat{\beta}_{\mathrm{PLS}}(k)\right] + \bar{y}^{\mathrm{ini}}\mathbb{1}.$$

Si nous obtenons une nouvelle valeur $x'_{n+1} = (x_{n+1,1}, \cdots, x_{n+1,p})$, il faut d'abord la centrer et la réduire avec les valeurs des moyennes et des écarts-types empiriques utilisées pour centrer et réduire les variables du tableau initial puis calculer

$$\hat{y}^{p}_{\mathrm{PLS},n+1}(k) = \hat{\sigma}_Y \sum_{j=1}^{p} \left(\frac{x_{n+1,j} - \bar{x}_j}{\hat{\sigma}_{X_j}} \left[\hat{\beta}_{\mathrm{PLS}}(k)\right]_j \right) + \bar{y}.$$

Au contraire de la régression (MC ou MCG), *l'estimateur de la régression PLS n'est pas une fonction linéaire de* Y. En effet, la prévision ne peut pas être mise sous la forme $\hat{Y}(k) = AY$ où A serait une matrice non dépendante de Y.
Une propriété notable de PLS est que $\forall k$, $\|\hat{\beta}_{\text{PLS}}(k)\| \leq \|\hat{\beta}\|$, où $\hat{\beta}$ est l'estimateur des MC. De plus, la norme $\|\hat{\beta}_{\text{PLS}}(k)\|$ augmente avec k (De Jong, 1995).

9.2.4 La régression PLS en pratique

Nous reprenons les données `ozone` :

```
> ozone <- read.table("ozone.txt",header=TRUE,sep=";",row.names=1)
```

Le code est quasiment identique à celui écrit pour **pcr**

```
> library(pls)
> set.seed(87)
> cvseg <- cvsegments(nrow(ozone), k = 4, type = "random")
> n.app <- nrow(ozone)
### modeles
> modele.pls <- plsr(O3 ~ ., ncomp=9, data = ozone[,1:10],scale=T,
+                  validation = "CV", segments = cvseg)
> msepcv.pls <- MSEP(modele.pls ,estimate=c("train","CV"))
> msepcv.pls
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps
train        559.8    173.9    150.9    146.9    144.2    142.1
CV           582.9    251.9    248.4    245.3    234.6    241.7
       6 comps  7 comps  8 comps  9 comps
train    141.4    140.9    139.8    139.7
CV       243.6    234.7    239.6    239.4
```

La régression PLS est conduite simplement grâce à la fonction **plsr**. Ici nous pouvons avoir au maximum 9 composantes principales ($\min(n_a, p) = p = 9$). La procédure de validation croisée 4 blocs choisit de retenir 4 composantes car l'erreur de prévision (notée CV) est la plus petite avec 4 composantes. Nous évaluons donc le modèle final.

```
> npls <- which.min(msepcv.pls$val["CV",,])-1
> modele.pls.fin <- plsr(O3~ . , ncomp  = npls, scale = TRUE,
+                     data = ozone[,1:10])
```

Les valeurs des coefficients sont obtenues dans le modèle centré réduit. Afin de retrouver les coefficients dans l'échelle initiale, il faut les diviser par les écarts-types respectifs, puis calculer l'intercept comme indiqué dans l'équation 9.6. Nous obtenons alors :

```
(Intercept)       T12       T15      Ne12       N12      S12
      63.08      0.42      0.53     -4.42      0.37     1.61
        E12       W12        Vx       O3v
       0.98      0.56      0.25      0.26
```

Le BIC de ce modèle avec 6 paramètres (4 coefficients pour les composantes sélectionnées, 1 coefficient pour l'intercept et 1 coefficient pour l'estimateur de σ) vaut 413.94. Le modèle obtenu par le package est donc très légèrement moins bon que le modèle trouvé en utilisant l'ordre des variables induit par $|\hat{\beta}_j^* \sqrt{\lambda_j}|$.

9.3 Exercices

Exercice 9.1 (Questions de cours)

1) La régression biaisée est en général utilisée lorsque l'hypothèse ci-dessous n'est pas satisfaite :
   A. $\mathcal{H}_1$ concernant le rang de X (matrice du plan d'expérience),
   B. $\mathcal{H}_2$ concernant l'espérance et la variance des résidus,
   C. $\mathcal{H}_3$ concernant la normalité des résidus.

2) Lorsque la matrice $(X'X)$ n'est pas inversible, l'estimateur des moindres carrés
   A. existe et est unique,
   B. existe et n'est pas unique,
   C. n'existe pas, aucun estimateur ne minimise les moindres carrés.

3) Lors d'une régression PCR, la première composante principale est la composante dont le produit scalaire avec Y est :
   A. maximum,
   B. minimum,
   C. aucun rapport.

4) Si $\mathrm{rang}(X) = p$, effectuer une régression PCR avec toutes les composantes donne les mêmes résultats qu'effectuer une régression MC classique :
   A. toujours,
   B. jamais,
   C. aucun rapport.

5) Nous pouvons calculer une régression PCR avec k composantes en effectuant k régressions univariées :
   A. faux,
   B. vrai,
   C. cela dépend des données.

6) Lors d'une régression PLS, la première composante PLS est la composante dont le produit scalaire avec Y est :
   A. maximum,
   B. minimum,
   C. aucun rapport.

7) Effectuer une régression PLS avec k composantes ou effectuer une régression PCR avec k composantes également donne les mêmes résultats :
   A. toujours,
   B. jamais,
   C. aucun rapport.

Exercice 9.2 (Régression sur composantes)
L'objectif de cet exercice est de proposer un code R qui fasse une régression sur composantes principales et choisisse ces composantes avec le BIC ou l'AIC.

Exercice 9.3 (Régression sur composantes)
Proposer un code R qui effectue une régression sur composantes principales et qui choisit les composantes avec les différents critères proposés ci-dessus.

Exercice 9.4 (Régression sur composantes)
L'objectif de cet exercice est de proposer un code R qui fasse une régression sur un nombre `nb` donné de composantes principales. Ces composantes sont prises en conservant l'ordre obtenu avec l'ACP.
Comparer vos résultats avec les résultats obtenus avec la fonction **pcr** du package **pls**. Attention, les coefficients donnés par la fonction **pcr** sont donnés pour les variables initiales centrées-réduites.

Exercice 9.5 (†Théorème 9.2)
Démontrer par récurrence le théorème 9.2 (indice : montrer aussi que $X^{(j)} = X \prod_{i=1}^{j-1}(I - w^{(i)}(t^{(i)\prime}t^{(i)})^{-1}t^{(i)\prime}X)$).

Exercice 9.6 (†Géométrie des estimateurs)
Soit les observations suivantes :

X_1	1	0	0
X_2	$1/\sqrt{3}$	$2/\sqrt{3}$	0
Y	1.5	0.5	1

Tableau 9.1 – Observations d'une régression.

Soit le modèle de régression multiple (sans constante) suivant :

$$Y = \beta_1 X_1 + \beta_2 X_2 + \varepsilon.$$

Les régressions ridge, lasso, PCR et PLS seront effectuées sur les variables sans centrage ni réduction.
1) Vérifier que $\Im(X) = (X)$ est le plan de $\mathbb{R}^3$ engendré par $\{\vec{i}, \vec{j}\}$.
2) Calculer $\hat{Y} = P_X Y$.
3) Représenter dans le plan $(\vec{i}, \vec{j})$ les points X_1, X_2 et $\hat{Y}$.
4) Que vaut p ici ? Représenter dans $\mathbb{R}^p$ l'ensemble B_1 des $\beta \in \mathbb{R}^p$ vérifiant la contrainte $\sum_{j=1}^p \beta_j^2 = \|\beta\|_2^2 = 1$. Faire de même avec B_2 l'ensemble des $\beta \in \mathbb{R}^p$ vérifiant la contrainte $\sum_{j=1}^p |\beta_j| = \|\beta\|_1 = 1$.
5) La matrice X peut être identifiée à une application linéaire de $\mathbb{R}^2$ dans $\mathbb{R}^3$. Donner intuitivement la forme des ensembles B_1 et B_2 lorsqu'on leur applique X (ellipse, cercle, parallélogramme...). Ces ensembles notés respectivement C_1 et C_2 sont définis par $C_1 = \left\{z \in \mathbb{R}^3, \exists \beta \in B_1 : z = X\beta\right\}$ et $C_2 = \left\{z \in \mathbb{R}^3, \exists \beta \in B_2 : z = X\beta\right\}$.
6) Vérifier grâce à un ordinateur que les formes de C_1 et C_2 données à la question précédente sont justes. Dessiner C_1 et C_2 sur le plan $(\vec{i}, \vec{j})$ de la question 3).
7) Représenter géométriquement $X\hat{\beta}_{\text{ridge}}$ et $X\hat{\beta}_{\text{lasso}}$ sur le plan précédent en utilisant C_1 et C_2 comme contraintes pour la régression ridge et lasso respectivement.

8) Que représente l'ensemble C_1 en termes de « composante » ? Trouver graphiquement la première composante PLS grâce à sa définition. Que représente l'ajustement de Y par la régression PLS à une composante, ajustement noté $\hat{Y}_{\text{PLS}}(1)$, en termes de projection de Y, c'est la projection de Y sur ... ? Représenter la réponse sur le graphique.
9) Calculer $X'X$, trouver le premier axe principal et en déduire la première composante principale.
10) Figurer la droite portée par la première composante principale X_1^* (géométriquement il s'agit du grand axe de C_1). Que représente $X_1^*\beta_1^*$ en termes de projection de Y, c'est la projection de Y sur ... ? Représenter la réponse sur le graphique.

Exercice 9.7 (Orthonormalisation)
L'objectif de cet exercice est de proposer un code R qui prenne en entrée une matrice centrée-réduite et retourne sa version orthonormalisée.

9.4 Notes

9.4.1 ACP et changement de base

L'ACP de X (ou du triplet $(X, I_p, I_n/n)$) peut être présentée comme la recherche d'un nouveau repère orthonormé pour les observations et d'une manière symétrique pour les variables mais nous ne parlerons pas de ce deuxième aspect. Lorsque l'on utilise l'ACP sur des données réelles, il est d'usage de centrer réduire au préalable le tableau X et c'est donc ce cas que nous envisagerons ici. Rappelons que les observations $i \in \{1, \ldots, n\}$ sont tout simplement les lignes du tableau de données X et sont donc des éléments de $\mathbb{R}^p$. Un nouveau repère orthonormé $a^1, a^2 \ldots$ va être choisi (par définition de l'ACP) afin que les coordonnées de tous les individus sur chaque axe soient les plus dispersées possibles. La suite d'axes va être construite en partant du premier : le premier axe a^1 est celui où les coordonnées sont le plus dispersées, le second est orthogonal au premier et tel que les coordonnées soient les plus dispersées (en sachant que le premier axe restreint la recherche), etc. La dispersion des coordonnées des individus i sur l'axe 1, ou inertie associée à l'axe 1, est calculée comme la somme des coordonnées : $\sum_{i=1}^n (\tilde{c}_i^1)^2/n$. Cet objectif (pour l'axe 1) s'écrit plus simplement comme $\|\tilde{c}^1\|^2/n$.
Avec cette définition, l'ACP revient à diagonaliser la matrice $\frac{1}{n}(X'X)$ et ses vecteurs propres normés à l'unité sont les axes (appelés axes principaux) $a^1, a^2 \ldots$ La matrice étant réelle de la forme $M'M$, elle est orthogonalement diagonalisable (car symétrique) et toutes ses valeurs propres sont positives ou nulles. La suite des axes orthonormés est donc obtenue en ordonnant les valeurs propres dans l'ordre décroissant $\tilde{\lambda}_1 > \tilde{\lambda}_2 > \ldots$ et en donnant les vecteurs propres associés $a^1, a^2 \ldots$. Dans la très grande majorité des cas pratiques, la matrice X est centrée réduite et la matrice $\frac{1}{n}(X'X)$ s'interprète alors comme la matrice des corrélations entre variables. Enfin, les vecteurs de coordonnées $\tilde{c}^1, \tilde{c}^2 \ldots$ sont appelés composantes principales normées à la valeur propre. Comme il s'agit de coordonnées sur des axes orthonormés, ces coordonnées sont calculées comme $\tilde{c}^1 = Xa^1, \tilde{c}^2 = Xa^2 \ldots$. Cela fait donc apparaître deux choses : les composantes principales sont des vecteurs de $\mathbb{R}^n$ donc de nouvelles variables, et elles sont construites comme combinaisons linéaires des variables de X. Enfin en considérant les produits scalaires $< \tilde{c}^i, \tilde{c}^j >$ et en remplaçant $\tilde{c}^i$ par sa définition Xa^i et avec l'information que les a^i sont des vecteurs propres orthonormés de $\frac{1}{n}(X'X)$, on en déduit que la famille des composantes principales $\tilde{c}^1, \tilde{c}^2 \ldots$ est une famille orthogonale et que les normes carrés de ces vecteurs valent $\tilde{\lambda}_i$, la valeur propre associée à l'axe i. En ACP, une étape de sélection du nombre

d'axes est intercalée avant l'analyse en limitant celle-ci aux axes porteurs d'information, c'est-à-dire ceux qui ont un grand pourcentage d'inertie : $\frac{\tilde{\lambda}_i}{\sum_{i=1} \tilde{\lambda}_i}$. Cette étape de choix des axes sera revisitée dans le cadre de la régression.
Si nous retournons au cadre de la régression, en supposant que les variables explicatives sont toutes centrées réduites, alors nous retombons bien sur le cadre envisagé en introduction : si nous diagonalisons $(X'X)$, nous obtenons une suite de vecteurs propres normés à l'unité $a^1, a^2 \ldots$ que nous pouvons ordonner en fonction des valeurs propres décroissantes $\lambda_1 \geq \lambda_2 \geq \ldots$ Ces valeurs propres sont simplement celles de l'ACP multipliées par n (i.e. $\lambda_i = n\tilde{\lambda}_i$). En construisant les composantes principales $\tilde{c}^1, \tilde{c}^2 \ldots$ grâce à $\tilde{c}^1 = Xa^1, \tilde{c}^2 = Xa^2 \ldots$ nous obtenons les nouvelles variables orthogonales que nous allons regrouper dans une matrice :

$$X^* = (\tilde{c}^1 | \tilde{c}^2 | \ldots | \tilde{c}^p).$$

Avec ces nouvelles variables explicatives X^*, nous souhaitons utiliser un modèle de régression (multiple, ridge, lasso, etc.) pour expliquer Y. Sachant que les variables de X^* sont issues du tableau centré réduit X, elles sont elles-mêmes centrées réduites. Il n'y a donc pas possibilité d'avoir autre chose qu'un modèle autour de 0 (les variables sont centrées), ce qui n'est pas le cas pour Y. Le modèle de régression doit donc ajouter un coefficient constant μ. Le modèle que l'on souhaiterait utiliser est donc le suivant

$$Y = \mu \mathbb{1} + X^* \beta^* + \varepsilon^*.$$

Ce modèle est en l'état peu praticable dans le sens où nous n'avons pas encore fait apparaître de lien avec les variables explicatives originelles et fait apparaître un nouveau vecteur d'erreur ε^*.
Rappelons que chaque colonne $\tilde{c}^j$ de X^* est construite grâce à $\tilde{c}^j = Xa^j$, nous en déduisons que matriciellement, pour toutes les colonnes,

$$X^* = XA,$$

où la matrice A contient les p axes principaux $a^1, \ldots, a^p$ rangé, en colonnes. Ensuite rappelons que la diagonalisation de $X'X$ donne matriciellement

$$X'X = A\Lambda A'$$

avec A la matrice orthogonale des axes principaux orthonormés ($A'A = AA' = I_p$) et $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_p)$ la matrice diagonale des valeurs propres ordonnées.
Notre modèle de régression classique (avec variables centrées réduites, d'où la constante μ) est le suivant

$$Y = \mu \mathbb{1} + X\beta + \varepsilon.$$

En utilisant $X^* = XA$ et le fait que $AA' = I_p$, nous obtenons

$$\begin{aligned} Y &= \mu \mathbb{1} + (XA)(A'\beta) + \varepsilon \\ Y &= \mu \mathbb{1} + X^* \beta^* + \varepsilon \end{aligned}$$

et nous avons obtenu une réécriture de notre modèle de régression classique en termes de variables explicatives orthogonales, les composantes principales.

9.4.2 Colinéarité parfaite : $|X'X| = 0$

Reprenons l'équation (9.1)

$$X'X = P\Lambda P'.$$

Le rang de X vaut maintenant k avec $k < p$, nous avons donc les $(p-k)$ dernières valeurs propres de $(X'X)$ qui valent zéro, $\lambda_{k+1} = \cdots = \lambda_p = 0$. Cela veut dire que pour tout $i > k$, nous avons

$$X_i^{*\prime} X_i^* = \lambda_i = 0. \tag{9.8}$$

Décomposons la matrice Λ en matrices blocs

$$\Lambda \quad = \quad \begin{pmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \Lambda_1 = \mathrm{diag}(\lambda_1, \ldots, \lambda_k),$$

et décomposons la matrice orthogonale P de taille $p \times p$ qui regroupe les vecteurs propres normés de $X'X$ en deux matrices P_1 et P_2 de taille respective $p \times k$ et $p \times (p-k)$. Soit $P = [P_1, P_2]$, nous avons alors

$$X^* = [X_1^*, X_2^*] = [XP_1, XP_2].$$

Cherchons maintenant la valeur de XP_2. Comme le rang de X vaut k, nous savons que la dimension de $\Im(X)$ vaut k et de même pour la dimension de $\Im(X'X)$. Ce sous-espace vectoriel possède une base à k vecteurs que l'on peut choisir orthonormés. Nous savons, par construction, que P_1 regroupe k vecteurs de base orthonormés de $\Im(X'X)$ tandis que P_2 regroupe $p-k$ vecteurs orthonormés (et orthogonaux aux k de P_1) qui complètent la base de $\Im(X'X)$ afin d'obtenir une base de $\mathbb{R}^p$. Nous avons donc que, quel que soit $u \in \Im(X'X)$, alors

$$u'P_2 \quad = \quad 0.$$

Prenons $u \neq 0$ et comme $u \in \Im(X'X)$, il existe $\gamma \in \mathbb{R}^p$ tel que $u = X'X\gamma \neq 0$. Nous avons donc

$$\gamma' X'XP_2 \quad = \quad 0,$$

pour tout $\gamma \in \mathbb{R}^p$ et donc $X'XP_2 = 0$, c'est-à-dire que $XP_2 = 0$. Nous avons alors

$$X^* = [X_1^*, X_2^*] = [XP_1, XP_2] = [XP_1, 0].$$

Au niveau des coefficients du modèle étoile, nous avons la partition suivante :

$$\beta^* = \begin{pmatrix} \beta_1^* \\ \beta_2^* \end{pmatrix} = \begin{pmatrix} P_1'\beta \\ P_2'\beta \end{pmatrix}.$$

Grâce à la reparamétrisation précédente, nous avons, avec $X_2^* = XP_2 = 0$,

$$\begin{aligned} Y &= X^*\beta^* + \varepsilon \\ &= X_1^*\beta_1^* + X_2^*\beta_2^* + \varepsilon \\ &= X_1^*\beta_1^* + \varepsilon. \end{aligned}$$

Cette paramétrisation nous assure donc que les moindres carrés dans le modèle initial et dans le modèle étoile sont égaux et nous allons donc utiliser le modèle étoile. Par les MC, nous obtenons $\hat{\beta}_1^* = (X_1^{*\prime} X_1^*)^{-1} X_1^{*\prime} Y$ et nous posons $\hat{\beta}_2^* = 0$, ce qui ne change rien car $X_2^* = 0$. Nous obtenons l'estimateur de la régression sur les k premières composantes principales (*principal component regression* : PCR)

$$\hat{\beta}_1^* = \Lambda_1^{-1} P_1' X' Y,$$

de variance

$$\mathrm{V}(\hat{\beta}_1^*) \quad = \quad \sigma^2 (X_1^{*\prime} X_1^*)^{-1} = \sigma^2 \Lambda_1^{-1}. \tag{9.9}$$

La stabilité des estimateurs peut être envisagée par leur variance, plus celle-ci est grande, plus l'estimateur sera instable. Cette variance dépend ici du bruit qui fait partie du problème et de λ_j. Une très faible valeur propre induit une grande variance et donc un estimateur instable et des conclusions peu fiables.
Nous avons donc que $\hat{\beta}_1^*$ minimise le critère des MC pour le modèle étoile. Comme les MC du modèle étoile et ceux du modèle initial sont égaux, à partir de $\hat{\beta}_1^*$, le vecteur des coefficients associés aux composantes principales, nous pouvons obtenir simplement $\hat{\beta}_{\mathrm{PCR}}$, le vecteur des coefficients associés aux variables initiales, par

$$\hat{\beta}_{\mathrm{PCR}} \quad = \quad P_1 \hat{\beta}_1^*.$$

Ce vecteur de coefficient minimise les MC du modèle initial. Le résultat est donc identique au paragraphe précédent à ceci près que l'on s'arrête aux k premières composantes principales associées aux valeurs propres non nulles de $(X'X)$.
Cela suggère le fait que l'on peut trouver une valeur, pour l'estimateur de la régression $\hat{\beta}$, qui est égale à $\hat{\beta}_1^*$. Mais nous pourrions trouver une infinité d'autres $\hat{\beta}$ qui seraient aussi solution de la minimisation des MC. Ils seraient tels que $\hat{\beta}_2^* \neq 0$. Cela donnerait une estimation $\hat{\beta} = P_1 \hat{\beta}_1^* + P_2 \hat{\beta}_2^*$. En plaçant cette valeur dans les moindres carrés, cela donne exactement les mêmes moindres carrés que ceux obtenus par $\hat{\beta}_{\mathrm{PCR}}$. Nous retrouvons là le fait que $\hat{\beta}$ n'est plus unique car $\mathcal{H}_1$ n'est plus vérifiée. En revanche, nous avons que $\hat{\beta}_{\mathrm{PCR}}$ est unique.
Puisque les résultats sont conservés quand l'on s'arrête à k, ce paragraphe suggère aussi que nous pouvons choisir une valeur de k de sorte que les valeurs propres associées $\{\lambda_j\}_{j=1}^k$ soient suffisamment différentes de 0, éliminant ainsi les problèmes de quasi non-inversibilité et de variance très grande. Evidemment, si l'on élimine les composantes principales associées à des valeurs propres non strictement nulles voire suffisamment grandes, la solution des MC dans le modèle initial et celle dans le modèle étoile seront différentes. Cependant, dans l'approche régression sur composantes principales, nous ne garderons que les estimateurs stables (*i.e.* de faible variance). Cette différence de moindres carrés est le prix à payer afin d'obtenir une solution unique et stable.

Chapitre 10

Comparaison des différentes méthodes, étude de cas réels

10.1 Erreur de prévision et validation croisée

Nous avons proposé dans les chapitres précédents plusieurs méthodes permettant d'expliquer une variable Y continue par p variables $X_1, \ldots, X_p$ (continues et/ou qualitatives). La question du choix de la « meilleure » méthode pour répondre à un problème de régression se pose alors naturellement. La notion de meilleure méthode, ou de meilleur modèle, nécessite de définir un critère qui permettra de comparer les différentes options. De nombreux critères ont déjà été proposés pour mesurer la performance d'une méthode. On peut par exemple citer l'AIC, le BIC, le C_p de Mallows (voir section 7.3). Ces critères, qui permettent notamment de choisir les variables dans un modèle linéaire, ne peuvent généralement pas être utilisés pour comparer toutes les méthodes (une régression linéaire avec une régression PLS par exemple). Dans ce cas, les approches classiques consistent à se baser sur des critères de prévision : on confronte les valeurs prédites par chaque méthode aux valeurs observées.

Nous rappelons que chaque méthode fournit un algorithme de prévision, cet algorithme est représenté par une fonction $m : \mathbb{R}^p \to \mathbb{R}$ qui, à une nouvelle observation $x \in \mathbb{R}^p$, renvoie une prévision $m(x) \in \mathbb{R}$. Les critères qui permettent de mesurer la performance d'un algorithme de prévision sont le plus souvent basés sur une fonction de perte $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$ telle que $\ell(y, m(x))$ mesure l'erreur ou le coût de la prévision $m(x)$ par rapport à l'observation y. En régression, on utilise souvent la perte quadratique $\ell(y, m(x)) = (y - m(x))^2$.

Nous rappelons qu'il n'est pas souhaitable d'utiliser les mêmes données pour estimer les paramètres du modèle et calculer l'erreur quadratique de prévision (voir section 7.2.4). On a souvent recours à des méthodes de ré-échantillonnage de type validation croisée dans ce cas-là. Ces techniques ont été présentées brièvement dans certaines parties précédentes. Nous rappelons ici les algorithmes d'apprentis-

sage/validation et de validation croisée dans un cadre général.

Algorithme 2 Apprentissage/Validation pour le calcul d'une erreur de prévision.

Entrées :
- les observations $(x_1, y_1), \ldots, (x_n, y_n)$;
- $\{\mathcal{A}, \mathcal{V}\}$ une partition de $\{1, \ldots, n\}$ en deux parties ;
- un algorithme de prévision ;
- une fonction de perte $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$.

1. Ajuster l'algorithme de prévision en utilisant uniquement les données d'apprentissage $\{(x_i, y_i) : i \in \mathcal{A}\}$. On désigne par $\widehat{m}$ l'algorithme obtenu.

2. Calculer la valeur prédite par l'algorithme pour chaque observation de l'échantillon de validation : $\widehat{m}(x_i), i \in \mathcal{V}$.

Retourner :

$$\frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \ell(y_i, \widehat{m}(x_i)).$$

Algorithme 3 Validation croisée K blocs pour le calcul d'une erreur de prévision.

Entrées :
- les observations $(x_1, y_1), \ldots, (x_n, y_n)$;
- $\{\mathcal{I}_1, \ldots, \mathcal{I}_K\}$ une partition de $\{1, \ldots, n\}$ en K blocs ;
- un algorithme de prévision ;
- une fonction de perte $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$.

Pour $k = 1, \ldots, K$:

1. Ajuster l'algorithme de prévision en utilisant l'ensemble des données privé du k^{e} bloc, c'est-à-dire $\{(x_i, y_i) : i \in \{1, \ldots, n\} \backslash \mathcal{I}_k\}$. On désigne par $\widehat{m}_k$ l'algorithme obtenu.

2. Calculer la valeur prédite par l'algorithme pour chaque observation du bloc k : $\widehat{m}_k(x_i), i \in \mathcal{I}_k$.

Retourner :

$$\frac{1}{n} \sum_{k=1}^{K} \sum_{i \in \mathcal{I}_k} \ell(y_i, \widehat{m}_k(x_i)).$$

L'approche apprentissage/validation consiste à séparer les données en deux : une partie pour calculer l'algorithme, une autre pour estimer sa performance. L'utilisateur doit choisir la séparation. L'échantillon d'apprentissage est souvent privilégié. En effet, cet échantillon est utilisé pour estimer les paramètres d'un modèle, par exemple les paramètres d'un modèle linéaire. Tandis que l'échantillon de validation

est utilisé pour estimer uniquement l'erreur de prévision. On prend souvent deux tiers ou trois quarts des observations dans l'échantillon d'apprentissage.
Pour la validation croisée, on doit séparer le jeu de données en K blocs, blocs généralement tirés au hasard. Ensuite, la procédure apprentissage/validation est répétée K fois en considérant chaque bloc comme échantillon de validation, les $K-1$ blocs restants constituent l'échantillon d'apprentissage.
Schématisons cette procédure pour une validation croisée $K = 4$ blocs en régression linéaire multiple. La figure 10.1 représente l'étape de séparation initiale.

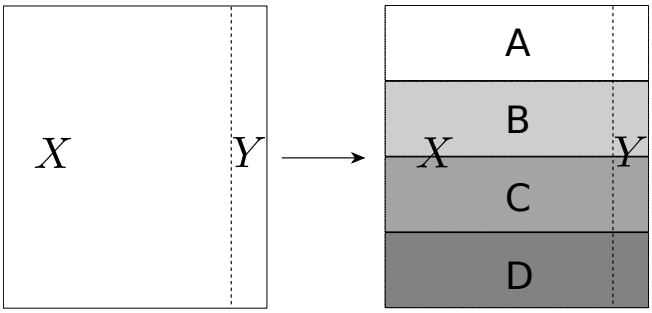


Fig. 10.1 – Validation croisée K blocs (avec $K = 4$) : étape initiale de séparation en blocs.

Une fois les blocs constitués, nous commençons par utiliser le bloc A en validation, les autres en apprentissage (voir fig. 10.2). Les blocs B, C et D sont donc utilisés pour estimer le vecteur β via une procédure de MCO. On obtient ainsi un $\hat{\beta}$ que l'on utilise pour prédire le Y des individus du bloc A : $\hat{y}_i = \hat{m}(x_i) = x_i'\hat{\beta}, i \in A$. On note $\hat{Y}$ le vecteur qui contient ces prévisions.

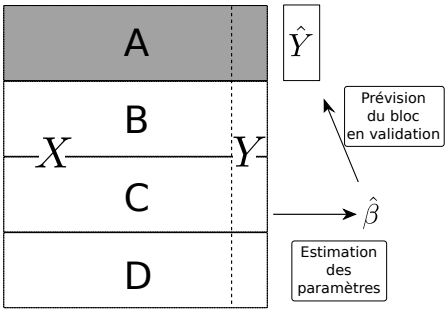


Fig. 10.2 – Validation croisée K blocs (avec $K = 4$, pour un modèle de régression) : seconde étape, le bloc A est utilisé en validation, les autres en apprentissage.

On répète ensuite cette procédure sur les blocs restants : on considère tout à tour les blocs B, C et D comme échantillon de validation afin d'obtenir une prévision pour tous les individus (voir fig. 10.3).

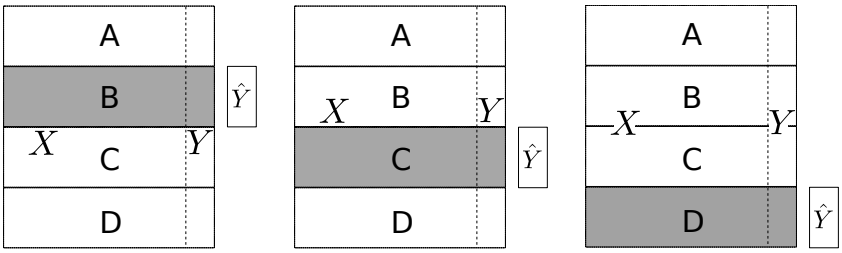


Fig. 10.3 – Validation croisée K blocs (avec $K = 4$) : étapes 3, 4 et 5, les blocs B, C et D sont respectivement utilisés en validation, les autres en apprentissage.

On évalue enfin la perte du modèle en confrontant les observations Y aux prévisions $\hat{Y}$. Pour la perte quadratique, on calculera ainsi

$$\frac{1}{n}\sum_{k=1}^{4}\sum_{i\in\mathcal{I}_k}(y_i - x_i'\hat{\beta})^2.$$

L'algorithme a été illustré pour $K = 4$ mais en pratique on utilise souvent $K = 10$. Lorsque $K = n$, on parle de validation croisée leave-one-out (l'abréviation loo est fréquemment utilisée). La validation croisée est souvent privilégiée par rapport à la méthode apprentissage/validation. Elle renvoie une prévision pour tous les individus et fournit donc une estimation de l'erreur plus stable. Elle est néanmoins plus coûteuse en temps de calcul car l'algorithme doit être calculé K fois.

Remarque

— Les algorithmes sont ici présentés pour une méthode de prévision. Si cette méthode nécessite un découpage des données, il faudra découper uniquement l'échantillon d'apprentissage (ou l'échantillon privé du k^{e} bloc pour la validation croisée). Si par exemple on souhaite faire de la régression lasso en sélectionnant le paramètre de régularisation par validation croisée, alors seul l'échantillon d'apprentissage (ou l'échantillon privé du k^{e} bloc) devra être découpé en bloc pour la sélection du paramètre.

— Lorsque la méthode de prévision dépend d'un paramètre, la valeur de ce paramètre ne sera pas forcément la même à chaque itération de la validation croisée. Par exemple, si on utilise une méthode de sélection pas à pas (voir section 7.4.2), les variables sélectionnées à chaque étape de la validation croisée ne seront pas forcément les mêmes. Les algorithmes 2 et 3 sont utilisés pour choisir une procédure (régression linéaire avec toutes les variables ou avec sélection de variable ou régression ridge/lasso...). Une fois la procédure choisie, on pourra la réajuster sur toutes les données pour obtenir le modèle final.

Nous proposons maintenant de comparer différentes méthodes pour l'exemple de l'ozone.

10.2 Analyse de l'ozone

Nous avons introduit de nombreuses méthodes basées sur la régression tout au long de ce livre et nous allons essayer de les comparer sur le jeu de données d'ozone afin d'obtenir la meilleur méthode de prévision de l'ozone.

10.2.1 Préliminaires

Commençons donc par importer le jeu de données complet :

```
> ozone <- read.table("ozone_complet.txt", header = T, sep = ";")
> dim(ozone)
[1] 1464   23
```

Ce jeu de données comporte 1464 observations et 22 variables potentiellement explicatives. Cependant des valeurs manquantes sont présentes, nous allons les identifier et les retirer.

```
> indNA <- which(is.na(ozone), arr.ind = T)[,1]
> ozone2 <- ozone[-indNA,]
```

Il reste 1366 observations. La variable à expliquer est `maxO3` et les variables potentiellement explicatives sont des données météorologiques à différentes heures de la journée (9 h, 12 h, 15 h et 18 h), le maximum de la veille `maxO3v`. La direction du vent est une variable « circulaire » : un vent à 0 degré est un vent du nord mais aussi à 360 degrés. Il semble plus logique alors de transformer vitesse et direction (qui sont les coordonnées polaires) en coordonnées cartésiennes afin d'obtenir `Vx` (axe est-ouest) et `Vy` (axe nord-sud). Le choix opéré est d'avoir `Vy` positif quand le vent est au Nord et `Vx` positif quand le vent est à l'est. Une fois ces transformations faites, la variable `Vx` qui était la projection du vent sur l'axe est-ouest à 12 h est redondante avec Vx12, elle est donc enlevée.
Afin de faciliter la répétabilité par le lecteur, nous fournissons les données transformées (fichier `ozone_transf.txt`) que nous importons

```
> ozone <- read.table("ozone_transf.txt", header = T, sep = ";")
```

donnant un jeu de données de 1366 individus et 22 variables (dont une à expliquer).

10.2.2 Méthodes et comparaison

Afin de comparer les différents modèles, nous allons appliquer la validation croisée avec 10 blocs. La présentation que nous allons faire sera un peu lourde (il va y avoir de fortes redondances de codes entre les méthodes) mais devrait permettre de bien comprendre le travail à faire.

Pour commencer, créons un data-frame qui va contenir les résultats des prévisions obtenues avec chaque méthode. Cet objet va donc admettre 1366 lignes qui correspondent alors aux 1366 observations des données initiales et autant de colonnes que de méthodes. Nous préconisons de rajouter une colonne dédiée à la valeur observée de `Y` ici le `maxO3`.

```
> RES <- data.frame(Y = ozone$maxO3)
```

Avant d'évaluer par validation croisée 10 blocs, nous effectuons un tirage aléatoire affectant chaque observation $i \in \{1, \ldots, n\}$ dans un bloc $b \in \{1, \ldots, 10\}$. Afin d'obtenir des blocs dont les effectifs sont équilibrés, nous effectuons une répartition séquentielle de chaque observation dans les blocs de la manière suivante : la première observation va dans le bloc 1, la seconde dans le bloc 2, ... la 10e dans le bloc 10, la 11e dans le bloc 1, la 12e dans le bloc 2 ...

```
> nbbloc <- 10
> blocseq <- rep(1:nbbloc, length = nrow(ozone))
```

Ensuite nous permutons aléatoirement l'appartenance aux blocs, ce qui nous donne donc un tirage aléatoire des blocs avec des effectifs équilibrés (résultat dans `bloc`) :

```
> set.seed(1234)
> bloc <- sample(blocseq)
```

Les méthodes de régression que nous allons comparer sur ces données seront la régression multiple, la régression multiple avec choix de variable selon un critère BIC, la régression lasso, ridge ou elastic-net et enfin deux types de régression sur composantes : PLS et PCR.

Régression multiple

Commençons par mettre en œuvre la régression multiple et la régression avec choix de variables par BIC (avec un algorithme exhaustif), nous utilisons les ordres R présentés à la section 7.5, p. 180. En préalable, chargeons la librairie **leaps** :

```
library(leaps)
for(i in 1:nbbloc){
  ###MCO global
  reg <- lm(maxO3~.,data=ozone[bloc!=i,])
  RES[bloc==i,"MCO"] <- predict(reg,ozone[bloc==i,])
  ###MCO choix
  recherche <- regsubsets(maxO3~., int=T, nbest=1, nvmax=22,
                                        data=ozone[bloc!=i,])
  resume <- summary(recherche)
  nomselec <- colnames(resume$which)[
                        resume$which[which.min(resume$bic),] ][-1]
```

```
  formule <- formula(paste("maxO3~",paste(nomselec,collapse="+")))
  regbic <- lm(formule,data=ozone[bloc!=i,])
  RES[bloc==i,"choix"] <- predict(regbic,ozone[bloc==i,])
}
```

Nous avons donc mis en pratique la régression et la régression avec choix de variables. Les résultats des prévisions sont dans le data-frame `RES`.

Attention, la procédure de choix de variables avec beaucoup de variables et la méthode `exhaustive` peuvent être coûteuses en temps de calculs, nous préconisons alors de remplacer l'option par défaut `method=exhaustive` par `method=forward` par exemple.

Afin de quantifier l'erreur de prévision que nous faisons, nous pouvons calculer l'erreur quadratique moyenne qui vaut 188.7 pour les MCO et 189.9 pour les MCO avec choix de variables. Nous pouvons maintenant répéter ce procédé pour les autres méthodes.

Lasso, ridge et elastic-net

Si nous poursuivons ces comparaisons, nous pouvons utiliser la procédure lasso pour effectuer une régression sous contraintes (voir section 8.3.1, p. 199). En préalable, rappelons que **glmnet** ne travaille qu'avec des matrices. La fonction **model.matrix** peut être utilisée pour construire la matrice des `X`. Cette fonction permet de plus d'effectuer le codage des variables explicatives qualitatives.

```
ozone.X <- model.matrix(maxO3~.,data=ozone)[,-1]
ozone.Y <- ozone[,"maxO3"]
```

La première colonne de **model.matrix** représente le coefficient constant (une colonne de 1) et est traitée séparément par **glmnet** ; nous l'éliminons donc de la matrice `X`. Pour chaque étape de validation croisée, il faut choisir le λ optimal et cela est fait grâce la fonction **cv.glmnet**. Voyons comment effectuer ce code

```
library(glmnet)
for(i in 1:nbbloc){
  XA <- ozone.X[bloc!=i,]
  YA <- ozone.Y[bloc!=i]
  XT <- ozone.X[bloc==i,]
  ###ridge
  rech <- cv.glmnet(XA,YA,alpha=0)
  RES[bloc==i,"ridge"] <- predict(rech, XT, s=rech$lambda.min)
```

```
  ###lasso
  rech <- cv.glmnet(XA,YA,alpha=1)
  RES[bloc==i,"lasso"] <- predict(rech, XT, s=rech$lambda.min)
  ###elastic
  rech <- cv.glmnet(XA,YA,alpha=0.5)
  RES[bloc==i,"elastic"] <- predict(rech, XT, s=rech$lambda.min)
}
```

Nous calculons maintenant l'erreur quadratique de prévision et obtenons

```
    MCO   choix   ridge   lasso elastic
  187.3   188.8   187.8   186.9   187.0
```

Régressions sur composantes

Dans cette section, nous allons implémenter les techniques vues au chapitre 9. Il est possible d'utiliser les fonctions proposées en exercice du chapitre 9 ou celles du package **pls**. Rappelons que, pour ces algorithmes, il est nécessaire de trouver le nombre de composantes (principales ou PLS). Les fonctions **plsr** et **pcr** choisissent ce nombre par validation croisée. L'utilisateur doit renseigner le nombre de composantes maximum, nous choisissons ici 20.

```
library(pls)
for(i in 1:nbbloc){
   #####PLS
   tmp <- plsr(maxO3~.,data=ozone[bloc!=i,],ncomp=20,
                                validation="CV",scale=TRUE)
   mse <- MSEP(tmp,estimate=c("train","CV"))
   npls <- which.min(mse$val["CV",,])-1
   mod <- plsr(maxO3~.,ncomp=npls,data=ozone[bloc!=i,],scale=TRUE)
   RES[bloc==i,"PLS"] <- predict(mod,ozone[bloc==i,],ncomp=npls)
   #####PCR
   tmp <- pcr(maxO3~.,data=ozone[bloc!=i,],ncomp=20,
                                   validation="CV",scale=TRUE)
   mse <- MSEP(tmp,estimate=c("train","CV"))
   npcr <- which.min(mse$val["CV",,])-1
   mod <- pcr(maxO3~.,ncomp=npcr,data=ozone[bloc!=i,],scale=TRUE)
   RES[bloc==i,"PCR"] <- predict(mod,ozone[bloc==i,],ncomp=npcr)
 }
```

Les résultats deviennent maintenant

```
    MCO   choix   ridge   lasso elastic     PLS     PCR
  187.3   188.8   187.8   186.9   187.0   187.3   187.3
```

Les MCO donnent de bons résultats comparés aux autres méthodes mais nous avons 1366 individus pour 22 variables. Les résultats risquent d'être différents si nous créons ou modifions des variables (features ingeniering). De plus, nous remarquons que les résultats obtenus sont très proches. Ces erreurs dépendent bien entendu du découpage en 10 blocs effectué. Afin d'obtenir des résultats plus stables, il est courant de répéter cette procédure de validation croisée sur plusieurs découpages en blocs et de faire la moyenne des résultats obtenus. Dans la section suivante, nous proposons donc de considérer des transformations des variables explicatives dans les modèles et de répéter la validation croisée plusieurs fois.

10.2.3 Pour aller plus loin

Dans la section précédente, nous avons considéré des modèles linéaires appliqués sur les variables explicatives brutes. Dit autrement, nous avons supposé que ces variables agissaient de façon linéaire sur la concentration en ozone. Il est bien entendu possible que cela ne soit pas vrai et que l'effet des variables explicatives soit quadratique ou autre. Cet effet est bien entendu difficile à trouver en pratique. Dans cette partie, nous proposons d'intégrer des effets polynomiaux et des interactions. De plus, nous répèterons les algorithmes de validation croisée plusieurs fois afin de stabiliser les erreurs calculées.

Afin de gagner en clarté de code, nous allons proposer une fonction qui, pour un bloc b donné, i) estime le modèle sur toutes les données sauf le bloc b, ii) calcule la prévision donnée par ce modèle sur les données du bloc b, iii) évalue la somme des écarts quadratiques entre observations et prévisions sur les données du bloc b, iv) et enfin en retour, la fonction donne la somme des écarts quadratiques.

Pour la régression multiple, cela donne :

```
sse_reg <- function(don, bloc, b) {
    m_reg <- lm(maxO3~., data=don[bloc!=b, ])
    previsions <- predict(m_reg, don[bloc==b, ])
    return(sum((don[bloc==b, "maxO3"] - previsions)^2))
}
```

Exercice 10.1 (Fonctions R)
En vous inspirant du code précédent et des codes déjà proposés, écrivez

1. une fonction **sse_regbic** qui prendrait en argument don, bloc, le nombre de variables maximales et le type d'algorithme et qui calculerait l'erreur de prévision obtenue sur les blocs par le modèle choisi ;

2. une fonction **sse_glmnet** qui prendrait en argument don, bloc et $alpha$ et qui calculerait l'erreur de prévision obtenue sur les blocs en utilisant ridge, ou lasso ou elastic-net (en fonction de α) en ayant choisi le paramètre optimal par la fonction **cv.glmnet** ;

3. une fonction **sse_pls** qui prendrait en argument don, bloc et le nombre de composantes maximales et qui calculerait l'erreur de prévision obtenue sur les blocs en utilisant la fonction **plsr** en ayant choisi le nombre de composants optimal ;

4. une fonction **sse_pcr** qui prendrait en argument don, bloc et le nombre de composantes maximales et qui calculerait l'erreur de prévision obtenue sur les blocs en utilisant la fonction **pcr** en ayant choisi le nombre de composants optimal.

Nous pouvons donc évaluer la régression multiple en répétant la procédure de validation croisée 10 blocs, 20 fois par exemple en exécutant le code suivant :

```
set.seed(1234)
nbbloc <- 10
ssereg  <- rep(0,20)
for (r in 1:20) {
  bloc <- sample(1:nbbloc,nrow(ozone),replace=TRUE)
  for(b in 1:nbbloc){
    ssereg[r] <- ssereg[r] + sse_reg(ozone,bloc,b)
  }
}
round(mean(ssereg/nrow(ozone)),2)
[1] 188.45
```

En faisant tourner tous les modèles précédents 20 fois, nous obtenons

```
   MCO    choix   ridge   lasso  elastic    PLS     PCR
188.36    189.6  188.66  187.92   187.82 188.27   188.46
```

Les modèles donnent quasiment tous les mêmes résultats. Essayons d'améliorer le modèle. La façon la plus simple consiste à introduire les interactions entre les variables explicatives via les produits 2 à 2 des variables.

Modèle de prévision avec interactions

Observons qu'il suffit de changer `maxO3~.` par `maxO3~.^2` à chaque fois où cela intervient. Afin de conserver les codes tels quels nous proposons de faire

```
ozone <- read.table("ozone_transf.txt",header=T,sep=";")
X <-  model.matrix(maxO3~.^2,data=ozone)[,-1]
ozone <- data.frame(maxO3=ozone$maxO3,X)
```

Nous obtenons ainsi 213 variables et nous pouvons exécuter la validation croisée répétée de la section précédente, en prenant soin de changer dans la fonction **ssei_regbic** la procédure exhaustive par une procédure forward (utiliser `method="forward"` [1] et le nombre de composantes maximales pour **plsr** et **pcr**. En faisant tourner tous les modèles précédents 20 fois, nous obtenons :

```
   MCO  choix  ridge  lasso elastic    PLS    PCR
185.04 166.24 165.82 162.12  162.31 172.35 173.99
```

1. Avec plus 50 variables explicatives, l'algorithme exhaustif est déconseillé.

La méthode elastic net donne la meilleure erreur de prévision. De plus, nous remarquons que, pour la plupart des méthodes, l'ordre de grandeur des erreurs a fortement diminué.

Modèle de prévision avec des polynômes

Nous pouvons facilement construire la matrice (ou un data-frame) avec comme variables, les variables initiales, les variables initiales élevées au carré, les variables initiales élevées au cube en faisant

```
ozone <- read.table("ozone_transf.txt",header=T,sep=";")
X <-  model.matrix(maxO3~.,data=ozone)[,-1]
ozone <- data.frame(maxO3=ozone$maxO3,X,X^2,X^3)
```

Nous avons 64 variables explicatives et les résultats sont

```
   MCO  choix  ridge  lasso elastic    PLS    PCR
166.34 168.81 166.04 164.95  165.20 168.13 167.36
```

Il est intéressant de noter qu'avec le modèle polynomial, toutes les méthodes font quasiment la même erreur.

Modèle de prévision avec des splines

Nous allons voir au chapitre 16 des fonctions splines. Les codes seront expliqués au chapitre 16 mais il nous semble intéressant de les mettre en application ici.

```
X <-  model.matrix(maxO3~.,data=ozone)[,-1]
library(splines)
BB <- NULL
for(i in 1:ncol(X)){
    var <- X[,i]
    BX <- bs(var,knots=quantile(var,prob=c(.25,.5,.75)),degre=3,
             Boundary.knots=c(min(var),max(var)))
    colnames(BX) <- paste(colnames(X)[i],"-b",1:6,sep="")
    BB <- cbind(BB,BX)
}
ozone <- data.frame(maxO3=ozone$maxO3,BB)
```

Nous avons 127 variables explicatives et les résultats sont

```
  MCO  choix   ridge  lasso elastic    PLS    PCR
165.95 162.97 158.96 156.49  156.39 162.67 164.67
```

Nous améliorons un petit peu les résultats précédents. Terminons ce chapitre avec des splines et de l'interaction.

Modèle de prévision avec des splines et de l'interaction

Nous obtenons les données en exécutant le code suivant

```
X <-  model.matrix(maxO3~.,data=ozone)[,-1]
library(splines)
BB <- NULL
for(i in 1:ncol(X)){
    var <- X[,i]
    BX <- bs(var,knots=quantile(var,prob=c(.25,.5,.75)),degre=3,
             Boundary.knots=c(min(var),max(var)))
    colnames(BX) <- paste(colnames(X)[i],"-b",1:6,sep="")
    BB <- cbind(BB,BX)
}
```

Nous allons concaténer les colonnes de BB et de la matrice de l'interaction mais dans cette dernière il y a les variables initiales qui sont également dans la matrice BB, il faut donc faire attention à ne pas prendre toutes les colonnes

```
X <-  model.matrix(maxO3~.^2,data=ozone)[,-1]
ozone <- data.frame(maxO3=ozone$maxO3,BB,X[,-c(1:21)])
```

Nous avons maintenant 337 variables explicatives et les résultats sont

```
   MCO  choix  ridge  lasso elastic    PLS    PCR
197.44 157.22 155.98 153.92  153.41 159.75 160.27
```

10.2.4 Conclusion

Il est intéressant de noter les meilleurs résultats sont de l'ordre de 155 de MSE alors que nous étions à 188 initialement. Et cela est atteint avec presque toutes les méthodes. Remarquons de plus que le choix des transformations des variables explicatives a un apport important dans la performance. A titre indicatif, nous avons fait tourner une forêt aléatoire avec les données initiales et nous avons trouvé un MSE de 155.0.

Quatrième partie

Le modèle linéaire généralisé

Chapitre 11

Régression logistique

Dans les chapitres précédents, nous avons essentiellement étudié le modèle de régression linéaire qui permet d'expliquer une variable quantitative continue. Nous présentons dans ce chapitre l'équivalent de ce modèle dans le cas où la variable à expliquer Y est qualitative et admet deux modalités. La plupart des notions seront identiques à celles présentées initialement. Nous ne les reprendrons donc pas toutes en détail et nous nous focaliserons sur les différences entre les deux modélisations.

11.1 Présentation du modèle

11.1.1 Exemple introductif

Nous reprenons le jeu de données présenté dans Hosmer & Lemeshow (2000) où le problème est d'expliquer la présence ou l'absence d'une maladie (variable `chd` qui vaut 1 si la pathologie est présente, 0 sinon) par l'âge du patient. On dispose de 100 individus, le tableau suivant donne les 5 premières observations.

Individu	`age`	`chd`
1	20	0
2	23	0
3	24	0
4	25	0
5	25	1

Tableau 11.1 – 5 observations de patients.

Le problème mettant en jeu simplement deux variables, il est possible de représenter la variable cible (`chd`) en fonction de la variable explicative (`age`) :

```
> artere <- read.table("chd.csv", sep = ";", header = TRUE)
> plot(chd ~ age, data = artere, pch = 16)
```

Chaque point du graphique (fig. 11.1) représente l'âge d'un patient en abscisse et la présence (ou l'absence) de la maladie en ordonnée.

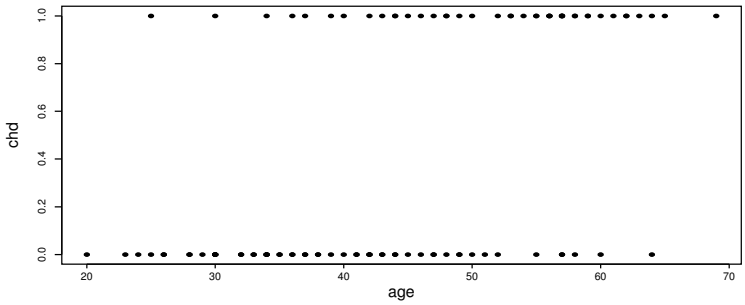


Fig. 11.1 – Variable `chd` en fonction de l'âge.

Une première solution pour traiter ce nouveau problème où la variable à expliquer est qualitative consisterait à coder `chd` en $\{0, 1\}$ (comme c'est fait dans les données brutes) ou $\{-1, 1\}$ et d'effectuer une régression linéaire. Une telle approche n'est clairement pas satisfaisante car le résultat dépendra du codage effectué. De plus, les valeurs ajustées et prédites par un modèle linéaire sont à valeurs dans $\mathbb{R}$ alors que les valeurs initiales sont dans un ensemble discret. Il faut donc proposer une alternative plus pertinente.

11.1.2 Modélisation statistique

Nous cherchons à modéliser une variable binaire Y (`chd`) en fonction des valeurs prises par une variable quantitative X (`age`). On va donc s'intéresser à la loi de Y pour chaque valeur que X peut prendre. Une approche naturelle consiste à modéliser cette loi par une loi de Bernoulli de paramètre $p \in [0, 1]$ inconnue. Une variable de Bernoulli étant à valeurs $\{0, 1\}$, on suppose sans perte de généralités que Y prend ses valeurs dans $\{0, 1\}$. Afin de prendre en compte l'âge du patient pour expliquer la variable Y, nous allons faire varier le paramètre de la loi de Bernoulli avec l'âge. Ainsi, pour un individu d'âge x fixé, Y va suivre une loi de Bernoulli de paramètre $p(x)$ où $p(x)$ désigne la probabilité de l'événement $\{Y = 1\}$ (être atteint par la maladie) pour un individu d'âge x. Il reste à spécifier une classe de fonctions pour $p(x)$.

On voit sur le graphe 11.1 que la maladie a tendance à être plus présente pour les personnes âgées. Afin de mieux quantifier ce lien et de se donner une idée plus précise de l'allure de la fonction $p(x)$, nous regardons dans le tableau 11.2 l'évolution de la proportion ou fréquence de personnes atteintes par la pathologie sur différentes classes d'âge.

		chd		
age	Effectifs	0	1	fréquence
]19 ;29]	10	9	1	0.1
]29 ;34]	15	13	2	0.133
]34 ;39]	12	9	3	0.25
]39 ;44]	15	10	5	0.333
]44 ;49]	13	7	6	0.462
]49 ;54]	8	3	5	0.625
]54 ;59]	17	4	13	0.765
]59 ;69]	10	2	8	0.8

Tableau 11.2 – Données regroupées en classe d'âge.

La liaison entre l'âge et la présence de la maladie devient beaucoup plus lisible. Il apparaît en effet que lorsque l'âge augmente, la proportion d'individus atteints par la maladie augmente comme on peut le voir sur le graphique suivant :

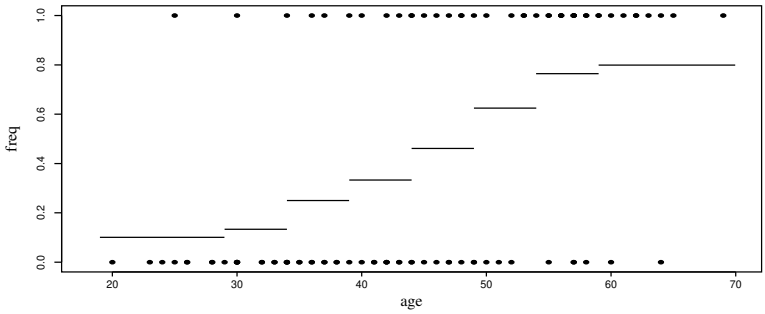


Fig. 11.2 – Fréquence de la variable `chd` par classe d'âge.

Cette fonction en escaliers dépend du regroupement effectué et peut donc changer en fonction des classes choisies. Par construction, il y a des sauts à chaque regroupement alors que l'on peut imaginer que la proportion évolue de façon continue avec l'âge. Afin de lisser la fonction en escaliers, il est d'usage d'utiliser une fonction de type *sigmoïde*, par exemple

$$\frac{\exp(\beta_1 + \beta_2 x)}{1 + \exp(\beta_1 + \beta_2 x)}$$

où le paramètre $\beta = (\beta_1, \beta_2) \in \mathbb{R}^2$ devra être choisi de manière à ajuster au mieux la fonction en escaliers (voir fig. 11.3).
Cette représentation suggère de modéliser la probabilité $p(x)$ d'être atteint par la maladie pour un patient d'âge x par la fonction $p_\beta(x)$ définie par

$$p_\beta(x) = \frac{\exp(\beta_1 + \beta_2 x)}{1 + \exp(\beta_1 + \beta_2 x)}. \tag{11.1}$$

Cette équation définit le *modèle logistique* dans le cas d'une variable explicative, on parle de modèle logistique simple. Le vecteur $\beta = (\beta_1, \beta_2) \in \mathbb{R}^2$ contient les paramètres inconnus du modèle. La fonction $u \mapsto \exp(u)/(1+\exp(u))$ étant bijective, le modèle (11.1) se réécrit

$$\mathsf{logit}(p_\beta(x)) = \log\left(\frac{p_\beta(x)}{1-p_\beta(x)}\right) = \beta_1 + \beta_2 x \tag{11.2}$$

où la fonction logit est définie sur $]0,1[$ par $u \mapsto \log(u/(1-u))$. La fonction **glm** de R permet d'estimer un modèle logistique :

```
> glm(chd ~ age, data = artere, family = binomial)

Call:  glm(formula = chd ~ age, family = binomial, data = artere)

Coefficients:
(Intercept)          age
    -5.3095       0.1109

Degrees of Freedom: 99 Total (i.e. Null);  98 Residual
Null Deviance:            136.7
Residual Deviance: 107.4         AIC: 111.4
```

On obtient ici $\hat{\beta}_1 = -5.3095$ et $\hat{\beta}_2 = 0.1109$. Ainsi, pour un individu d'âge x, on estimera avec ce modèle que la probabilité d'avoir la maladie à l'âge x est donnée par

$$p_{\hat{\beta}}(x) = \frac{\exp(-5.3095 + 0.1109x)}{1+\exp(-5.3095+0.1109x)}.$$

Cette fonction est représentée en pointillés sur la figure 11.3. On voit clairement que le paramètre β a été estimé de manière à bien ajuster la courbe en escaliers.

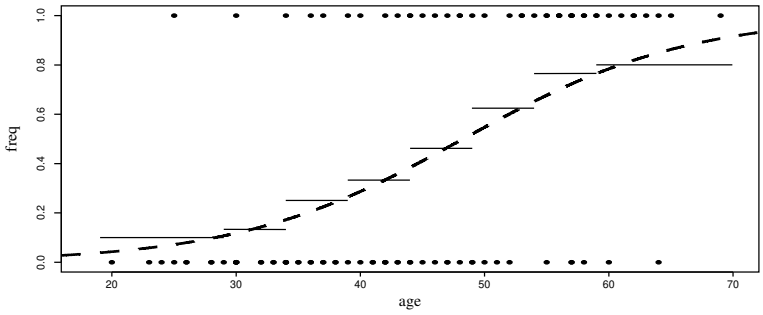


Fig. 11.3 – Fréquence de la variable `chd` par classe d'âge.

Le modèle défini en (11.1) et (11.2) permet d'expliquer une variable binaire Y par une seule variable X. Il se généralise directement au cas où on est confronté à p variables explicatives $X_1, \ldots, X_p$.

Définition 11.1
Soit $(x_1, y_1), \ldots, (x_n, y_n)$ *n observations telles que* $x_i = (x_{i1}, \ldots, x_{ip}) \in \mathbb{R}^p$ *et* $y_i \in \{0, 1\}$. *Le modèle de régression logistique suppose que les observations* $y_i, i = 1, \ldots, n$ *sont des réalisations de variables aléatoires binaires indépendantes et de loi de Bernoulli de paramètre* $p_\beta(x_i)$ *vérifiant*

$$\mathsf{logit}(p_\beta(x_i)) = \log\left(\frac{p_\beta(x_i)}{1 - p_\beta(x_i)}\right) = \beta_1 x_{i1} + \ldots + \beta_p x_{ip} = x_i'\beta. \tag{11.3}$$

Pour une valeur de x fixée, ce modèle fait le lien entre la probabilité de l'événement $\{Y = 1\}$ (modulo la transformation logit) et la combinaison linéaire des variables explicatives $x'\beta$.

Remarque
Tout comme pour le modèle linéaire, si on souhaite inclure la variable constante dans le modèle, on prendra x_{i1} égal à 1 pour tout i variant de 1 à n.

11.1.3 Variables explicatives qualitatives, interactions

La nature des variables explicatives (quantitatives ou qualitatives) intervient uniquement dans l'écriture de la combinaison linéaire $\beta_1 x_1 + \ldots + \beta_p x_p$ du modèle logistique. En présence d'une seule variable explicative quantitative X, le modèle logistique (avec constante) va s'écrire

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 x.$$

Le coefficient β_1 mesure l'effet de la variable X sur la probabilité que Y soit égale à 1. Par exemple, une valeur de β_1 positive signifiera que $p_\beta(x)$ augmente lorsque x augmente.
Si maintenant la variable X est qualitative, alors l'écriture $\beta_0 + \beta_1 x$ n'a plus de sens. Considérons le cas où X prend ses valeurs dans $\{A, B, C\}$, la combinaison linéaire $\beta_0 + \beta_1 x$ ne veut plus rien dire puisque x vaut A, B ou C. La solution est identique à ce qui est fait dans le modèle linéaire (voir chapitre 6) : on code la variable qualitative à l'aide d'indicatrices

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x).$$

Rappelons que pour un ensemble E donné, la fonction $\mathbf{1}_E(x)$ vaut 1 si $x \in E$, 0 sinon. Nous sommes à nouveau confrontés à des problèmes d'identifiabilité similaires à ceux abordés au chapitre 6. En effet, comme $\mathbf{1}_A(x) + \mathbf{1}_B(x) + \mathbf{1}_C(x) = 1$, n'importe quelle indicatrice peut s'écrire en fonction des deux autres. Cela entraîne que différentes valeurs de $\beta = (\beta_0, \ldots, \beta_3)$ renvoient la même loi pour Y. On a par exemple

$$\begin{aligned}\mathsf{logit}(p_\beta(x)) &= \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x) \\ &= 0 + (\beta_0 + \beta_1)\mathbf{1}_A(x) + (\beta_0 + \beta_2)\mathbf{1}_B(x) + (\beta_0 + \beta_3)\mathbf{1}_C(x).\end{aligned}$$

Ce modèle n'est donc pas identifiable et il devient nécessaire de fixer des *contraintes identifiantes*. Les contraintes les plus courantes consistent à fixer à 0 le coefficient associé à une modalité, appelée *modalité de référence*. Dans ce cas, si on fixe par exemple A comme modalité de référence, le modèle se réécrit

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x)$$

muni de la contrainte $\beta_1 = 0$, que l'on peut réécrire

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x). \tag{11.4}$$

Une autre contrainte classique consiste à annuler la somme des coefficients associés à la variable qualitative :

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x)$$

muni de la contrainte $\beta_1 + \beta_2 + \beta_3 = 0$. Ce qui peut se réécrire

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + (-\beta_1 - \beta_2) \mathbf{1}_C(x). \tag{11.5}$$

Exemple 11.1
On cherche à expliquer une variable binaire Y par une variable qualitative X à valeurs dans $\{A, B, C\}$. On génère un échantillon de taille 100 tel que la loi de X est uniforme sur $\{A, B, C\}$ et la loi conditionnelle de Y sachant X est définie par

$$\text{Loi}(Y|X = A) = \text{Loi}(Y|X = C) = \mathcal{B}(0.9) \quad \text{et} \quad \text{Loi}(Y|X = B) = \mathcal{B}(0.1). \tag{11.6}$$

L'échantillon $(x_i, y_i), i = 1, \ldots, 100$ est obtenu avec les commandes suivantes :

```
> set.seed(12345)
> X <- factor(sample(c("A","B","C"), 100, replace = T))
> Y <- rep(0, 100)
> Y[X=="A"] <- rbinom(sum(X=="A"),1,0.9)
> Y[X=="B"] <- rbinom(sum(X=="B"),1,0.1)
> Y[X=="C"] <- rbinom(sum(X=="C"),1,0.9)
> donnees <- data.frame(X,Y)
```

On ajuste un modèle logistique

```
> model <- glm(Y ~ ., data = donnees, family = binomial)
> coef(model)
(Intercept)          XB          XC
 1.68639895 -3.55820113 -0.01242252
```

Aucun coefficient n'est renvoyé pour la modalité A. Par défaut, la fonction **glm** prend comme modalité de référence la première des modalités rangées dans l'ordre lexicographique. Le modèle considéré est donc ici le modèle (11.4). On lit les estimations : $\hat{\beta}_0 = 1.68639895$, $\hat{\beta}_2 = -3.55820113$ et $\hat{\beta}_3 = -0.01242252$. Ces coefficients sont à interpréter en fonction de la contrainte : $\hat{\beta}_2$ étant négatif, on peut

dire que la probabilité que $Y = 1$ est plus faible lorsque X vaut B que lorsque X vaut A. $\hat{\beta}_3$ est proche de 0, cela signifie que lorsque X est égal à C la probabilité de l'événement $Y = 1$ est proche de la probabilité de ce même événement lorsque X vaut A. Ces interprétations sont cohérentes avec le procédé de simulation des observations (voir (11.6)).
Il est bien entendu possible de modifier la contrainte identifiante. Par exemple on ajustera le modèle (11.5) avec la commande

```
> model1 <- glm(Y ~ C(X,sum), data = donnees, family = binomial)
> coef(model1)
(Intercept)  C(X, sum)1  C(X, sum)2
  0.4961911   1.1902079  -2.3679932
```

On a alors comme estimation : $\hat{\beta}_0 = 0.4961911$, $\hat{\beta}_1 = 1.1902079$, $\hat{\beta}_2 = -2.3679932$ et $\hat{\beta}_3 = -\hat{\beta}_1 - \hat{\beta}_2 = 1.177785$. Bien entendu, la contrainte choisie n'a pas d'influence sur les valeurs prédites. Il est en effet facile de voir que les paramétrisations (11.4) et (11.5) renvoient les mêmes estimations des probabilités de l'évènement $\{Y = 1\}$ pour tout $x \in \{A, B, C\}$. Le choix de la paramétrisation (et donc de la contrainte) a en revanche une influence sur l'interprétation des paramètres (voir exercice 11.2).

Les interactions se traitent de la même façon que pour le modèle de régression linéaire. Nous renvoyons le lecteur au chapitre 6 ainsi qu'à l'exercice 11.5 pour plus de précisions.

11.2 Estimation

La nature qualitative de la variable à expliquer ne permet pas d'utiliser une approche du type moindres carrés pour estimer le vecteur β. Nous avons vu que l'estimateur des moindres carrés de β coïncide avec l'estimateur du maximum de vraisemblance (EMV) dans le modèle linéaire. C'est cette dernière méthode qui est généralement utilisée pour estimer les paramètres du modèle logistique.

11.2.1 La vraisemblance

On dispose de n observations $(x_1, y_1), \ldots, (x_n, y_n)$ où $(x_i, y_i) \in \mathbb{R}^p \times \{0, 1\}$ et les y_i sont des réalisations de variables aléatoires indépendantes de loi de Bernoulli de paramètre $p_\beta(x_i)$ vérifiant (11.3). La vraisemblance du modèle logistique s'écrit donc

$$L(Y, \beta) = \prod_{i=1}^{n} p_\beta(x_i)^{y_i} (1 - p_\beta(x_i))^{1-y_i}$$

avec $Y = (y_1, \ldots, y_n)$ et $\beta = (\beta_1, \ldots, \beta_p)$. Comme son nom l'indique, l'approche du maximum de vraisemblance consiste à maximiser en β la fonction $L(Y, \beta)$. Il

est souvent plus simple de considérer la log-vraisemblance

$$\begin{aligned}\mathcal{L}(Y,\beta) = \log L(Y,\beta) &= \sum_{i=1}^{n} \{y_i \log(p_\beta(x_i)) + (1-y_i)\log(1-p_\beta(x_i))\} \\ &= \sum_{i=1}^{n} \left\{ y_i \log\left(\frac{p_\beta(x_i)}{1-p_\beta(x_i)}\right) + \log(1-p_\beta(x_i)) \right\},\end{aligned}$$

ou encore d'après (11.3),

$$\mathcal{L}(Y,\beta) = \sum_{i=1}^{n} \{y_i x_i'\beta - \log(1+\exp(x_i'\beta))\}. \tag{11.7}$$

Si la vraisemblance admet un maximum fini, alors celui-ci annule le gradient de la log-vraisemblance défini par

$$\nabla\mathcal{L}(Y,\beta) = \left(\frac{\partial\mathcal{L}(Y,\beta)}{\partial\beta_1}, \ldots, \frac{\partial\mathcal{L}(Y,\beta)}{\partial\beta_p}\right).$$

Pour chaque composante du gradient, on a

$$\frac{\partial\mathcal{L}(Y,\beta)}{\partial\beta_j} = \sum_{i=1}^{n} \left[y_i x_{ij} - \frac{x_{ij}\exp(x_i'\beta)}{1+\exp(x_i'\beta)} \right] = \sum_{i=1}^{n} \left[x_{ij}(y_i - p_\beta(x_i)) \right], \quad j = 1, \ldots, p.$$

Par conséquent

$$\nabla\mathcal{L}(Y,\beta) = \sum_{i=1}^{n} \left[x_i(y_i - p_\beta(x_i)) \right] = X'(Y - P_\beta) \tag{11.8}$$

où

$$X = \begin{pmatrix} x_{11} & \ldots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{np} \end{pmatrix}, \quad Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{et} \quad P_\beta = \begin{pmatrix} p_\beta(x_1) \\ \vdots \\ p_\beta(x_n) \end{pmatrix}.$$

Le gradient $\nabla\mathcal{L}(Y,\beta)$, vu comme une fonction de β, est généralement appelé la *fonction de score* et est noté $S(\beta)$. Ainsi, s'il existe, l'estimateur du MV $\hat{\beta}$ est une solution de l'équation

$$S(\beta) = \nabla\mathcal{L}(Y,\beta) = X'(Y - P_\beta) = 0. \tag{11.9}$$

Résoudre cette équation revient à résoudre p équations à p inconnues $(\beta_1, \ldots, \beta_p)$:

$$x_{1j}y_1 + \ldots + x_{nj}y_n = x_{1j}\frac{\exp(x_1'\beta)}{1+\exp(x_1'\beta)} + \ldots + x_{nj}\frac{\exp(x_n'\beta)}{1+\exp(x_n'\beta)}, \; j = 1, \ldots, p.$$

Ce système non linéaire en β n'admet pas de solution analytique. Nous devons donc nous tourner vers des méthodes numériques pour pouvoir calculer l'estimateur du maximum de vraisemblance. Les méthodes numériques utilisées par la plupart des logiciels statistiques sont basées sur des algorithmes itératifs qui permettent de trouver les extrema locaux de fonctions. La proposition suivante justifie l'utilisation d'algorithmes itératifs pour calculer l'EMV.

Proposition 11.1
Supposons la matrice X de plein rang, alors la log-vraisemblance $\mathcal{L}(Y, \beta)$ est strictement concave par rapport à β.

La preuve de cette proposition est à faire dans l'exercice 11.4. La concavité a une conséquence importante : un algorithme itératif convergera vers l'estimateur du maximum du vraisemblance lorsque celui-ci existe. Elle ne garantit cependant pas l'existence d'une solution finie au problème de maximisation. Les cas où les solutions ne sont pas finies sont néanmoins rares (voir exercice 11.3).

11.2.2 Calcul des estimateurs : l'algorithme IRLS

Deux algorithmes sont généralement implémentés dans les logiciels pour calculer les estimateurs du maximum de vraisemblance dans les modèles linéaires généralisés : l'algorithme du score de Fisher et l'algorithme de Newton Raphson. Dans le cas du modèle logistique, ces deux algorithmes coïncident. Nous les présentons dans cette section.
L'approche « Newton Raphson » consiste à trouver une suite $(\beta^{(k)})_{k\in\mathbb{N}}$ de vecteurs de $\mathbb{R}^p$ qui converge vers l'estimateur du maximum de vraisemblance $\hat{\beta}$. On rappelle que, s'il existe, $\hat{\beta}$ est solution de l'équation de score (11.9) et vérifie donc

$$S(\hat{\beta}) = \nabla\mathcal{L}(Y, \hat{\beta}) = 0. \tag{11.10}$$

On cherche à construire la suite $(\beta^{(k)})_{k\in\mathbb{N}}$ par récurrence : il suffit donc de trouver une formule de récurrence permettant de calculer $\beta^{(k+1)}$ à partir de $\beta^{(k)}$. Soit $\beta^{(k)} \in \mathbb{R}^p$, un développement de Taylor à l'ordre 1 de la fonction de score donne l'approximation

$$0 = S(\hat{\beta}) \approx S(\beta^{(k)}) + A(\beta^{(k)})(\hat{\beta} - \beta^{(k)}). \tag{11.11}$$

$A(\beta^{(k)})$ désigne la matrice hessienne de la log-vraisemblance au point $\beta^{(k)}$ définie par

$$A(\beta^{(k)}) = \nabla^2\mathcal{L}(Y, \beta^{(k)}) = -X'W_{\beta^{(k)}}X$$

où W_β est la matrice $n \times n$ diagonale dont le i^{e} terme de la diagonale est défini par $p_\beta(x_i)(1 - p_\beta(x_i))$. Si X est de plein rang, alors $A(\beta^{(k)})$ est inversible. On obtient ainsi en résolvant (11.11)

$$\hat{\beta} \approx \beta^{(k)} - A^{-1}(\beta^{(k)})S(\beta^{(k)}).$$

D'où la formule de récurrence

$$\beta^{(k+1)} = \beta^{(k)} - A^{-1}(\beta^{(k)})S(\beta^{(k)})$$

de laquelle on déduit l'algorithme 4.

Algorithme 4 Maximisation de la vraisemblance (IRLS).

1. **Initialisation** : fixer $\beta^{(0)} \in \mathbb{R}^p$, $k = 1$;
2. **Répéter**

— $\beta^{(k+1)} \leftarrow \beta^{(k)} - A^{-1}(\beta^{(k)})S(\beta^{(k)})$
— $k \leftarrow k + 1$

jusqu'à $\beta^{(k+1)} \approx \beta^{(k)}$ et/ou $\mathcal{L}(Y, \beta^{(k+1)}) \approx \mathcal{L}(Y, \beta^{(k)})$.

La dernière ligne de l'algorithme correspond au critère d'arrêt du processus itératif. Le processus est généralement stoppé lorsque $\beta^{(k+1)}$ est très proche de $\beta^{(k)}$. La manière de mesurer la proximité peut varier d'un logiciel à l'autre. Par défaut, la fonction **glm** utilise comme critère d'arrêt la vraisemblance : l'algorithme est stoppé dès que

$$|\mathcal{L}(Y, \beta^{(k+1)}) - \mathcal{L}(Y, \beta^{(k)})| \leq \varepsilon$$

avec $\varepsilon > 0$ petit ou dès qu'un nombre maximal d'itérations a été atteint.

Remarque
En utilisant (11.8), la formule de récurrence de l'algorithme de maximisation peut se réécrire

$$\begin{aligned}
\beta^{(k+1)} &= \beta^{(k)} + (X'W_{\beta^{(k)}}X)^{-1}X'(Y - P_{\beta^{(k)}}) \\
&= (X'W_{\beta^{(k)}}X)^{-1}X'W_{\beta^{(k)}}(X\beta^{(k)} + {W_{\beta^{(k)}}}^{-1}(Y - P_{\beta^k}) \\
&= (X'W_{\beta^{(k)}}X)^{-1}X'W_{\beta^{(k)}}Z^{(k)},
\end{aligned}$$

où $Z^{(k)} = X\beta^{(k)} + {W_{\beta^{(k)}}}^{-1}(Y - P_{\beta^{(k)}})$. On déduit que $\beta^{(k+1)}$ s'obtient en effectuant la régression pondérée du vecteur $Z^{(k)}$ par la matrice X, d'où le nom de *Iterative Reweighted Least Square* (IRLS) pour cet algorithme. Les poids $W_{\beta^{(k)}}$ dépendent de X et $\beta^{(k)}$ et sont réévalués à chaque étape de l'algorithme.

11.2.3 Propriétés asymptotiques de l'EMV

Nous avons vu dans la partie précédente qu'il n'existe pas d'écriture explicite pour l'EMV. Il est donc impossible d'utiliser directement les résultats classiques tels que la loi des grands nombres ou le théorème central limite pour étudier le comportement asymptotique de l'EMV. Il est néanmoins bien connu que, sous certaines hypothèses de régularité, il existe des garanties théoriques pour l'EMV. Il est par exemple consistant et asymptotiquement normal avec comme matrice de variance covariance l'inverse de la matrice d'information de Fisher. Dans le cas du modèle logistique, cette matrice est donnée par

$$\mathcal{I}_n(\beta) = -\mathbb{E}\left[\nabla^2\mathcal{L}(Y, \beta)\right] = X'W_\beta X. \tag{11.12}$$

Le théorème suivant, dont on pourra trouver la preuve dans Fahrmeir & Kaufman (1985) et Antoniadis *et al.* (1992), énonce les principales propriétés asymptotiques de l'EMV.

Théorème 11.1
On suppose que :
— *les $x_i, i = 1, \ldots, n$ prennent leurs valeurs dans une partie compacte de $\mathbb{R}^p$;*
— *la plus petite valeur propre $\lambda_{min}(X'X)$ tend vers $+\infty$ lorsque $n \to \infty$;*
— *la vraie valeur de β est finie.*
Alors il existe une suite de variables aléatoires $(\hat{\beta}_n)_n$ telle que

1. $\lim_{n\to\infty} \Pr(S(\hat{\beta}_n) = 0) = 1$ *où S est la fonction de score définie par* (11.9).
2. $\hat{\beta}_n$ *converge en probabilité vers β lorsque n tend vers $+\infty$.*
3. $\hat{\beta}_n$ *est asymptotiquement gaussienne telle que :*

$$\mathcal{I}_n(\beta)^{1/2}(\hat{\beta}_n - \beta) \xrightarrow{\mathcal{L}} \mathcal{N}(0, I_p) \quad \textit{quand} \quad n \to \infty. \tag{11.13}$$

$\xrightarrow{\mathcal{L}}$ désigne la convergence en loi et I_p la matrice identité de dimension $p \times p$. La première assertion garantit que l'EMV existe, au moins pour n suffisamment grand. Les points 2 et 3 précisent que l'EMV va se rapprocher de la vraie valeur de β lorsque le nombre d'observations augmente. Le point 3 nous donne de plus la loi asymptotique de l'EMV, c'est de cette loi que sont déduits les intervalles de confiance et les procédures de tests sur les paramètres du modèle logistique. Afin de simplifier les notations, on désignera dans la suite l'estimateur du MV par $\hat{\beta}$ (on supprime l'indice n).

Remarque
La compacité de l'espace des régresseurs n'est pas une hypothèse restrictive. En pratique, les valeurs des variables explicatives varient le plus souvent dans une partie compacte de $\mathbb{R}^p$. La seconde hypothèse implique que l'information (au sens de Fisher) sur le paramètre β augmente lorsque le nombre d'observations tend vers $+\infty$. Elle est nécessaire pour augmenter la précision (en termes de diminution de la variance) de $\hat{\beta}$ lorsque le nombre d'observations n augmente.

11.3 Intervalles de confiance et tests

Le calcul d'intervalles de confiance ainsi que le développement de procédures permettant de tester la valeur des paramètres du modèle se font à partir de la normalité asymptotique des estimateurs du MV présentée dans le théorème 11.1. Il faudra donc prendre garde d'avoir suffisamment d'observations pour utiliser ces procédures. On peut réécrire la convergence en loi (11.13) :

$$(\hat{\beta} - \beta)'\mathcal{I}_n(\beta)(\hat{\beta} - \beta) \xrightarrow{\mathcal{L}} \chi^2_p,$$

où on rappelle que χ_p désigne la loi du chi-2 à p degrés de liberté. La matrice d'information $\mathcal{I}_n(\beta)$ dépendant du paramètre inconnu β, on ne peut pas utiliser ce

résultat pour calculer des intervalles de confiance pour les paramètres β_j. Cependant, $\hat{\beta}$ étant consistant, on déduit des opérations classiques sur la convergence en loi que

$$(\hat{\beta} - \beta)'\mathcal{I}_n(\hat{\beta})(\hat{\beta} - \beta) \xrightarrow{\mathcal{L}} \chi^2_p. \tag{11.14}$$

Ainsi pour n grand, on pourra approcher la loi de $\hat{\beta}$ par une loi gaussienne multivariée centrée en β et de matrice de variance covariance $(X'W_{\hat{\beta}}X)^{-1}$.

11.3.1 IC et tests sur les paramètres du modèle

Les intervalles de confiance ainsi que les tests sur les paramètres du modèle logistique peuvent s'effectuer à partir de la propriété suivante qui est une conséquence directe de (11.14).

Propriété 11.1
Sous les hypothèses du théorème 11.1, on a

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma}_j} \xrightarrow{\mathcal{L}} \mathcal{N}(0,1), \tag{11.15}$$

où $\hat{\sigma}_j^2$ désigne le j^e terme de la diagonale $(X'W_{\hat{\beta}}X)^{-1}$.

Un intervalle de confiance (asymptotique) de niveau $1-\alpha$ pour β_j est donc donné par

$$IC_{1-\alpha}(\beta_j) = \left[\hat{\beta}_j - u_{1-\alpha/2}\hat{\sigma}_j; \hat{\beta}_j + u_{1-\alpha/2}\hat{\sigma}_j\right], \tag{11.16}$$

où $u_{1-\alpha/2}$ représente le quantile d'ordre $(1-\alpha/2)$ de la loi normale $\mathcal{N}(0,1)$.
La proposition 11.1 permet également d'effectuer des tests de nullité des coefficients du modèle. On note $\mathrm{H}_0 : \beta_j = 0$ et $\mathrm{H}_1 : \beta_j \neq 0$. Sous H_0, on déduit de (11.15) que $\hat{\beta}_j/\hat{\sigma}_j \xrightarrow{\mathcal{L}} \mathcal{N}(0,1)$. On rejettera H_0 au niveau $\alpha \in]0,1[$ si la valeur observée de $\hat{\beta}_j/\hat{\sigma}_j$ dépasse en valeur absolue le quantile d'ordre $1-\alpha/2$ de la loi $\mathcal{N}(0,1)$.

Exemple 11.2
On considère le jeu de données `SAheart` du package **bestglm**.

```
> library(bestglm)
> data(SAheart)
```

Tout comme pour les données présentées en introduction de ce chapitre, il s'agit d'expliquer la présence ou l'absence d'une maladie (variable `chd`). Afin de pouvoir effectuer de la prévision sur de nouvelles données (limitées à 3 pour des questions de lisibilité) nous les mettons de côté

```
new.SAheart <- SAheart[c(2, 408, 35), ]
row.names(new.SAheart) <- NULL
SAheart <- SAheart[-c(2,408,35), ]
```

On dispose ici de 9 variables explicatives et de 459 individus (et 3 à prévoir). On ajuste sous R le modèle logistique permettant d'expliquer `chd` par les autres variables du jeu de données.

```
> model <- glm(chd ~ ., data = SAheart, family = binomial)
> round(summary(model)$coefficients,4)
(Intercept)     -6.0837     1.3141 -4.6294   0.0000
sbp              0.0065     0.0058  1.1268   0.2598
tobacco          0.0814     0.0269  3.0232   0.0025
ldl              0.1794     0.0600  2.9891   0.0028
adiposity        0.0184     0.0295  0.6224   0.5337
famhistPresent   0.9325     0.2291  4.0694   0.0000
typea            0.0392     0.0123  3.1845   0.0015
obesity         -0.0637     0.0446 -1.4300   0.1527
alcohol          0.0002     0.0045  0.0346   0.9724
age              0.0439     0.0122  3.5923   0.0003
```

Comme pour la régression linéaire, on obtient une matrice qui comporte pour chaque paramètre (chaque ligne) 4 colonnes :

— son estimation $\hat{\beta}_j$ (colonne `Estimate`) ;
— son écart-type estimé $\hat{\sigma}_j$ (colonne `Std.Error`) ;
— la valeur observée de la statistique du test d'hypothèse $\mathrm{H}_0 : \beta_j = 0$ contre $\mathrm{H}_1 : \beta_j \neq 0$, c'est-à-dire $\hat{\beta}_j/\hat{\sigma}_j$ (colonne `z value`) ;
— la probabilité critique du test (colonne `Pr(>|z|)`).

On remarque qu'au niveau $\alpha = 5\%$, on acceptera la nullité de 4 coefficients (`sbp, adiposity, obesity,` et `alcohol`). On obtient les intervalles de confiance (11.16) à l'aide de la fonction **confint.default** :

```
> confint.default(model)
                      2.5 %       97.5 %
(Intercept)    -8.659355064 -3.507983650
sbp            -0.004797560  0.017773140
tobacco         0.028628110  0.134174033
ldl             0.061770934  0.297043348
adiposity      -0.039461469  0.076187468
famhistPresent  0.483354369  1.381572764
typea           0.015090098  0.063396115
obesity        -0.151054913  0.023612482
alcohol        -0.008639577  0.008950096
age             0.019931097  0.067793230
```

La fonction **confint** permet d'obtenir également des intervalles de confiance mais ces derniers sont calculés à partir de la vraisemblance (voir exercice 11.9 pour plus de détails).

11.3.2 Test sur un sous-ensemble de paramètres

La propriété 11.1 permet de tester la nullité d'un paramètre dans le modèle logistique. Il existe de nombreuses situations où on souhaite tester la nullité d'un sous-ensemble de coefficients du modèle. Considérons par exemple le scénario où on souhaite expliquer une variable Y binaire par une variable X_1 qualitative à trois modalités (A, B, C) et deux variables X_2 et X_3 continues. Le modèle s'écrit

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x_1) + \beta_2 \mathbf{1}_B(x_1) + \beta_3 \mathbf{1}_C(x_1) + \beta_4 x_2 + \beta_5 x_3, \quad (11.17)$$

muni de la contrainte $\beta_1 = 0$. Plusieurs situations peuvent motiver la construction de tests de nullité d'un sous-ensemble de paramètres. Nous citons quelques exemples ci-dessous.

— **Mesure de l'effet d'une variable qualitative dans le modèle** : il s'agit de tester la nullité de l'ensemble des coefficients associés aux modalités de la variable en question. Pour mesurer l'effet de X_1 dans le modèle (11.17), on testera

$$\mathrm{H}_0 : \beta_1 = \beta_2 = \beta_3 \quad \text{contre} \quad \mathrm{H}_1 : \exists (j,k) \in \{1,2,3\}^2 \text{ tel que } \beta_j \neq \beta_k.$$

Compte tenu de la contrainte identifiante, les hypothèses se réécrivent :

$$\mathrm{H}_0 : \beta_2 = \beta_3 = 0 \quad \text{contre} \quad \mathrm{H}_1 : \exists j \in \{2,3\} \text{ tel que } \beta_j \neq 0.$$

— **Test sur la validité globale du modèle** : on teste si au moins un coefficient, à l'exception de la constante, est non nul (équivalent du test de Fisher global en régression linéaire). Pour le modèle (11.17), il s'agira donc de tester

$$\mathrm{H}_0 : \beta_2 = \ldots = \beta_5 = 0 \quad \text{contre} \quad \mathrm{H}_1 : \exists j \in \{2,\ldots,5\} \text{ tel que } \beta_j \neq 0.$$

— **Tests entre modèles emboîtés** : on veut choisir un modèle parmi deux modèles emboîtés. On veut par exemple comparer le modèle (11.17) au modèle logistique permettant d'expliquer Y par X_1 uniquement. Cela revient à tester dans (11.17) les hypothèses

$$\mathrm{H}_0 : \beta_4 = \beta_5 = 0 \quad \text{contre} \quad \mathrm{H}_1 : \exists j \in \{4,5\} \text{ tel que } \beta_j \neq 0.$$

On se place dans le cas général où on cherche à tester la nullité de $q < p$ paramètres dans le modèle

$$\mathsf{logit}(p_\beta(x)) = \beta_1 x_1 + \ldots + \beta_p x_p.$$

Sans perte de généralité, on supposera que l'on veut tester la nullité des q premiers paramètres :

$$\mathrm{H}_0 : \beta_1 = \ldots = \beta_q = 0 \quad \text{contre} \quad \mathrm{H}_1 : \exists j \in \{1,\ldots,q\} \text{ tel que } \beta_j \neq 0.$$

On note $\hat{\beta} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)$ l'estimateur du MV de β et $\hat{\beta}_{1:q} = (\hat{\beta}_1, \ldots, \hat{\beta}_q)$ le vecteur comprenant les q premières coordonnées de $\hat{\beta}$. D'après (11.14), $\hat{\beta}_{1:q}$ est, pour n assez grand, gaussien d'espérance $\beta_{1:q} = (\beta_1, \ldots, \beta_q)$ et de matrice de variance-covariance égale au bloc supérieur gauche de dimension $q \times q$ de

$$\mathcal{I}_n(\hat{\beta})^{-1} = (X'W_{\hat{\beta}}X)^{-1}.$$

On note $\mathcal{I}_{1:q}(\hat{\beta})^{-1}$ cette matrice. On déduit du théorème 11.1

$$(\hat{\beta}_{1:q} - \beta_{1:q})'\mathcal{I}_{1:q}(\hat{\beta})(\hat{\beta}_{1:q} - \beta_{1:q}) \xrightarrow{\mathcal{L}} \chi^2_q,$$

où $\mathcal{I}_{1:q}(\hat{\beta})$ désigne l'inverse de $\mathcal{I}_{1:q}(\hat{\beta})^{-1}$. On déduit que sous H_0

$$(\hat{\beta}_{1:q})'\mathcal{I}_{1:q}(\hat{\beta})(\hat{\beta}_{1:q}) \xrightarrow{\mathcal{L}} \chi^2_q,$$

et on rejettera H_0 si la valeur observée de la statistique ci-dessus dépasse le quantile d'ordre $1 - \alpha$ de la loi χ^2_q. Ce test est appelé *test de Wald*.

Remarque
Les estimateurs des paramètres du modèle logistique étant calculés en maximisant la vraisemblance, on peut aussi utiliser les statistiques de *rapport de vraisemblance* pour tester la nullité d'un ou plusieurs paramètres. En conservant les notations de ci-dessus, on a sous H_0

$$-2(\mathcal{L}_{\mathrm{H}_0}(Y, \hat{\beta}_{\mathrm{H}_0}) - \mathcal{L}(Y, \hat{\beta})) \xrightarrow{\mathcal{L}} \chi^2_q$$

où $\mathcal{L}_{\mathrm{H}_0}(Y, \hat{\beta}_{\mathrm{H}_0})$ désigne la log-vraisemblance du modèle sous H_0, c'est-à-dire la log-vraisemblance du modèle

$$\mathsf{logit}(p_\beta(x)) = \beta_{q+1}x_{q+1} + \ldots + \beta_p x_p,$$

et $\hat{\beta}_{\mathrm{H}_0}$ est l'EMV des paramètres de ce modèle.

Exemple 11.3
On souhaite expliquer une variable Y binaire par une variable X_1 qualitative à 3 modalités (A, B, C) et deux variables continues X_2 et X_3. Le modèle logistique s'écrit

$$\mathsf{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x_1) + \beta_2 \mathbf{1}_B(x_1) + \beta_3 \mathbf{1}_C(x_1) + \beta_4 x_2 + \beta_5 x_3, \quad (11.18)$$

muni de la contrainte $\beta_1 = 0$. On génère $n = 1000$ observations :

```
> n <- 1000
> set.seed(123)
> X1 <- sample(c("A","B","C"), n, replace = TRUE)
> X2 <- rnorm(n)
> X3 <- runif(n)
> cl <- 1+0*(X1=="A")+1*(X1=="B")-3*(X1=="C")+2*X2
> Y <- rbinom(n, 1, exp(cl)/(1+exp(cl)))
> donnees <- data.frame(X1,X2,X3,Y)
```

On estime les paramètres β_j avec

```
> m1 <- glm(Y ~ ., data = donnees, family = binomial)
```

La fonction **Anova** du package **car** permet d'effectuer les tests de Wald et du rapport de vraisemblance pour tester l'influence des variables dans les modèles. Cette fonction permettra donc de tester la nullité des paramètres associés à chaque variable :

— X_1 : $\mathrm{H}_0 : \beta_2 = \beta_3 = 0$ contre $\mathrm{H}_1 : \beta_2 \neq 0$ ou $\beta_3 \neq 0$;
— X_2 : $\mathrm{H}_0 : \beta_4 = 0$ contre $\mathrm{H}_1 : \beta_4 \neq 0$.
— X_3 : $\mathrm{H}_0 : \beta_5 = 0$ contre $\mathrm{H}_1 : \beta_5 \neq 0$.

On effectue d'abord les tests de Wald.

```
> library(car)
> Anova(m1, type = 3, test.statistic = "Wald")
Response: Y
            Df    Chisq Pr(>Chisq)
(Intercept)  1  28.4698  9.517e-08 ***
X1           2 212.5061  < 2.2e-16 ***
X2           1 210.3902  < 2.2e-16 ***
X3           1   0.3096     0.5779
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Pour les tests du rapport de vraisemblance, il faut modifier l'option `test.statistic`

```
> Anova(m1, type = 3, test.statistic = "LR")
Analysis of Deviance Table (Type III tests)

Response: Y
   LR Chisq Df Pr(>Chisq)
X1   376.74  2     <2e-16 ***
X2   417.66  1     <2e-16 ***
X3     0.31  1     0.5778
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Dans les deux cas, l'hypothèse nulle est rejetée pour X_1 et X_2 et ne l'est pas pour X_3. Si l'utilisateur ne souhaite pas utiliser de package spécialisé, la base de R permet faire les tests du rapport de vraisemblance en utilisant des tests entre modèles emboîtés. En effet, tester l'influence de la variable X_1 dans le modèle revient à tester le modèle (11.18) contre ce modèle sans la variable X_1. L'utilisateur doit alors construire les modèles sans chacune des trois variables :

```
> m01 <- glm(Y ~ X2 + X3, data=donnees, family=binomial)
> m02 <- glm(Y ~ X1 + X3, data=donnees, family=binomial)
> m03 <- glm(Y ~ X1 + X2, data=donnees, family=binomial)
```

Ces modèles sont ensuite testés contre le modèle complet (appelé `model`). Le test permettant de regarder l'influence de la variable X_1 est conduit ainsi :

```
> anova(m01, m1, test = "LRT")
Analysis of Deviance Table

Model 1: Y ~ X2 + X3
Model 2: Y ~ X1 + X2 + X3
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       997    1109.67
2       995     732.93  2   376.74 < 2.2e-16 ***
```

et nous concluons au seuil de 5 % que la variable X_1 est utile. Pour X_2 et X_3, on utilisera

```
> anova(m02, m1, test = "LRT")
> anova(m03, m1, test = "LRT")
```

Nous retrouvons bien évidemment les mêmes résultats que pour le test du rapport de vraisemblance avec la fonction **Anova** du package **car**.
Pour les tests de Wald, on pourra également utiliser la fonction **wald.test** du package **aod**. On retrouvera par exemple le test pour l'utilité de la variable X_1 avec

```
> library(aod)
> wald.test(Sigma = vcov(m1), b = coef(m1), Terms = c(2,3))
Wald test:
----------

Chi-squared test:
X2 = 212.5, df = 2, P(> X2) = 0.0
```

11.3.3 Prévision

Pour le modèle logistique, le concept de prévision est légèrement différent de celui que nous avons vu pour le modèle linéaire. En régression linéaire, l'objectif est de prédire (ou d'estimer) $y_{n+1} = x'_{n+1}\beta$ pour une nouvelle observation x_{n+1}. Pour le modèle logistique, on peut de la même manière se poser la question de prédire le label ou groupe $y_{n+1} \in \{0, 1\}$ d'une nouvelle observation $x_{n+1} \in \mathbb{R}^p$. Cette question n'a cependant pas de sens à ce stade, puisque ce label n'est pas clairement défini. La quantité définie est la probabilité $p_\beta(x_{n+1})$. C'est à partir de

cette probabilité qu'on peut définir un label en posant

$$y_{n+1}(s) = \begin{cases} 1 & \text{si } p_\beta(x_{n+1}) \geq s \\ 0 & \text{sinon} \end{cases} \tag{11.19}$$

où s est un seuil fixé dans $[0, 1]$. Le seuil de 0.5 paraît bien entendu être le plus naturel. Il n'est cependant pas le plus opportun dans certaines situations. Nous reviendrons sur ce problème de choix de seuil dans le chapitre 14.
Etant donné une nouvelle observation $x_{n+1} \in \mathbb{R}^p$ et un seuil $s \in [0, 1]$, on cherche donc à prédire le groupe $y_{n+1}(s)$ défini par (11.19). On pose naturellement

$$\hat{y}_{n+1}(s) = \begin{cases} 1 & \text{si } p_{\hat{\beta}}(x_{n+1}) \geq s \\ 0 & \text{sinon,} \end{cases}$$

où $\hat{\beta}$ désigne l'estimateur du maximum de vraisemblance de β. On prédira ainsi que l'individu x_{n+1} est dans le groupe 1 si la probabilité

$$p_{\hat{\beta}}(x_{n+1}) = \frac{\exp(x'_{n+1}\hat{\beta})}{1 + \exp(x'_{n+1}\hat{\beta})}$$

est supérieure au seuil fixé s.
La quantité ci-dessus est un estimateur de $p_\beta(x_{n+1})$, la probabilité sous le modèle logistique que x_{n+1} soit dans le groupe 1. Dans de nombreuses situations, il peut être intéressant de donner un intervalle de confiance pour cette probabilité. D'après le théorème 11.1, $\hat{\beta}$ est (pour n grand) un vecteur gaussien d'espérance β et de matrice de variance covariance $\mathcal{I}_n(\beta)^{-1}$. Par conséquent, la loi de $x'_{n+1}\hat{\beta}$ peut être approchée pour n grand par la loi gaussienne

$$\mathcal{N}(x'_{n+1}\beta, x'_{n+1}\mathcal{I}_n(\beta)^{-1}x_{n+1}).$$

En posant $\hat{\sigma}^2_{n+1} = x'_{n+1}(X'W_{\hat{\beta}}X)^{-1}x_{n+1}$, on déduit un intervalle de confiance asymptotique de niveau $1 - \alpha$ pour $p_\beta(x_{n+1})$:

$$\left[\frac{\exp(x'_{n+1}\hat{\beta} - u_{1-\alpha/2}\hat{\sigma}_{n+1})}{1 + \exp(x'_{n+1}\hat{\beta} - u_{1-\alpha/2}\hat{\sigma}_{n+1})} ; \frac{\exp(x'_{n+1}\hat{\beta} + u_{1-\alpha/2}\hat{\sigma}_{n+1})}{1 + \exp(x'_{n+1}\hat{\beta} + u_{1-\alpha/2}\hat{\sigma}_{n+1})} \right] \tag{11.20}$$

où $u_{1-\alpha/2}$ désigne le quantile d'ordre $1 - \alpha/2$ de la loi $\mathcal{N}(0, 1)$.

Exemple 11.4
On reprend le jeu de données `SAheart` sur lequel on a ajusté le modèle logistique suivant :

```
> model <- glm(chd ~ ., data = SAheart, family = binomial)
```

Rappelons que nous disposons de 3 nouvelles observations pour lesquelles on souhaite prévoir la variable à expliquer (`chd`) :

```
> new.SAheart
  sbp tobacco  ldl adiposity famhist typea obesity alcohol age chd
1 144    0.01 4.41     28.61  Absent    55   28.87    2.06  63   1
2 200   19.20 4.43     40.60 Present    55   32.04   36.00  60   1
3 148    5.50 7.10     25.31  Absent    56   29.84    3.60  48   0
```

La fonction **predict** permet de calculer des prévisions pour ces trois individus :

```
> predict(model, newdata = new.SAheart)
         1          2          3
-0.7496872  2.0036264 -0.5350967
```

Par défaut, cette fonction renvoie la valeur de la combinaison linéaire des variables explicatives prédites, c'est-à-dire $x'_{n+1}\hat{\beta}$. Si on désire obtenir les probabilités $p_{\hat{\beta}}(x_{n+1})$, il faut ajouter l'argument `type="response"` :

```
> predict(model, newdata = new.SAheart, type = "response")
        1         2         3
0.3208895 0.8811773 0.3693289
```

On pourra déduire de ces estimations les labels prédits pour ces 3 nouveaux individus. Si on considère un seuil de 0.5, on prédira `chd=1` pour le deuxième individu et `chd=0` pour les individus 1 et 3. On pourra enfin obtenir les intervalles de confiance (11.20) avec les commandes :

```
> prev <- predict(model, newdata=new.SAheart, type="link",
+                 se.fit = TRUE)
> cl_inf <- prev$fit - qnorm(0.975)*prev$se.fit
> cl_sup <- prev$fit + qnorm(0.975)*prev$se.fit
> binf <- exp(cl_inf) / (1  +exp(cl_inf))
> bsup <- exp(cl_sup) / (1 + exp(cl_sup))
> data.frame(binf,bsup)
       binf      bsup
1 0.1997174 0.4722004
2 0.7138812 0.9566007
3 0.2461812 0.5122202
```

11.4 Adéquation du modèle

Nous présentons dans cette partie des indicateurs qui permettent, dans une certaine mesure, de mesurer l'adéquation ou la qualité d'ajustement aux données du modèle logistique. Les critères d'adéquation sont le plus souvent construits en confrontant les données observées aux valeurs ajustées par le modèle ou en comparant le modèle à un modèle de référence qui ajuste bien les données. Nous

commençons par présenter le *modèle saturé* qui est souvent pris comme référence pour mesurer l'adéquation.

11.4.1 Le modèle saturé

Rappelons que nous sommes en présence de n observations $(x_1, y_1), \ldots, (x_n, y_n)$ où $x_i \in \mathbb{R}^p$ et $y_i \in \{0, 1\}$. Les y_i sont vus comme des réalisations de loi de Bernoulli de paramètre $p_i \in [0, 1]$. Dans ce contexte, poser un modèle revient à mettre une contrainte entre les $p_i, i = 1, \ldots, n$. Nous avons vu par exemple que le modèle logistique suppose $p_i = p_\beta(x_i)$ tel que (11.3) est vérifié. Par définition, le *modèle saturé* ne pose aucune contrainte sur les p_i. Il possède donc n paramètres $p_1, \ldots, p_n$ inconnus si tous les x_i sont différents. On a alors autant de paramètres à estimer que d'observations, d'où le terme saturé.

Lorsque les x_i ne sont pas tous différents, le nombre de paramètres à estimer diminue. En effet, si $x_i = x_j$ alors les probabilités de succès p_i et p_j sont identiques. Il suffit alors d'estimer une probabilité par point de design. On définit ainsi

— $\tilde{x}_1, \ldots, \tilde{x}_T, T \leq n$ l'ensemble des valeurs prises par $x_1, \ldots, x_n$;
— $n_t = \text{Card}\{i : x_i = \tilde{x}_t\}$, $t = 1, \ldots, T$;
— $\tilde{y}_t = \sum_{\{i : x_i = \tilde{x}_t\}} y_i$.

T correspond au nombre de points de design différents, $\tilde{x}_t$ représente un point du design, n_t désigne le nombre de fois où on a observé le point $\tilde{x}_t$ et $\tilde{y}_t$ le nombre de succès observé au point $\tilde{x}_t$. Avec cette formalisation $\tilde{y}_t$ est la réalisation d'une loi binomiale $\mathcal{B}(n_t, p_t)$. Cette nouvelle écriture généralise le modèle considéré au préalable. En effet, si tous les $x_i, i = 1, \ldots, n$ sont différents, alors $T = n$, $\tilde{x}_t = x_t$, $n_t = 1$ et $\tilde{y}_t = y_t$.

Exemple 11.5
On considère un jeu de 10 observations avec deux variables explicatives X_1 et X_2. Les tableaux 11.3 et 11.4 représentent les données aux formats individuel et répété.

i	x_1	x_2	y
1	3	2	0
2	1	4	1
3	2	4	1
4	1	4	0
5	3	2	0
6	1	4	1
7	2	4	1
8	1	4	1
9	3	2	1
10	1	3	0

Tableau 11.3 – Format individuel.

t	$\tilde{x}_1$	$\tilde{x}_2$	n_t	$\tilde{y}$
1	3	2	3	1
2	1	4	4	3
3	2	4	2	2
4	1	3	1	0

Tableau 11.4 – Format répété.

Comme nous l'avons dit précédemment, le modèle saturé ne pose pas de contraintes

entre les probabilités p_t. La propriété suivante présente les estimateurs du maximum de vraisemblance dans ce modèle (voir exercice 11.8 pour la preuve).

Propriété 11.2
On considère le modèle saturé dans le cas de données répétées.

1. *La log-vraisemblance du modèle saturé en* $p = (p_1, \ldots, p_T) \in [0,1]^T$ *vaut*

$$\mathcal{L}_{sat}(Y,p) = \sum_{t=1}^{T} \left(\log \binom{n_t}{\tilde{y}_t} + \tilde{y}_t \log(p_t) + (n_t - \tilde{y}_t) \log(1 - p_t) \right)$$

2. *Les estimateurs du maximum de vraisemblance de* $p = (p_1, \ldots, p_T) \in [0,1]^T$ *sont donnés par*

$$\hat{p}_t = \frac{\tilde{y}_t}{n_t} \quad t = 1, \ldots, T.$$

En particulier, si tous les x_i *sont différents, on a* $\hat{p}_i = y_i, i = 1, \ldots, n$.

3. *Soit* $\mathcal{L}(Y,\beta)$ *la log-vraisemblance en* $\beta = (\beta_1, \ldots, \beta_p)$ *du modèle logistique*

$$\mathsf{logit}(p_\beta(x)) = \beta_1 x_1 + \ldots + \beta_p x_p.$$

Alors pour tout $\beta \in \mathbb{R}^p$

$$\mathcal{L}(Y,\beta) \leq \mathcal{L}_{sat}(Y,\hat{p}),$$

où $\hat{p} = (\hat{p}_1, \ldots, \hat{p}_T)$.

Cette propriété s'interprète naturellement : en l'absence de contraintes la probabilité que la variable cible soit égale à 1 au point $\tilde{x}_t$ est estimée par la proportion de succès observée au point $\tilde{x}_t$. Lorsqu'on a peu de répétitions pour les points du design, on peut ainsi s'attendre à une mauvaise qualité d'estimation des paramètres du modèle saturé. En effet la variance des $\hat{p}_t$ est grande lorsque n_t est petit. C'est pourquoi le modèle saturé est souvent sur-paramétré. C'est néanmoins un modèle qui sera souvent performant en termes d'ajustement. La propriété 3 nous dit en effet que c'est le « meilleur » modèle en termes de vraisemblance.

Exemple 11.6 (Modèle saturé pour l'exemple introductif (p. 257))
Reprenons les données de la maladie coronarienne (`chd`). Les points du design peuvent être obtenus simplement grâce à la fonction **unique** appliquée à la variable explicative (la fonction peut être utilisée avec une ou plusieurs variables explicatives) :

```
> unique(artere[,"age"])
[1] 20 23 24 25 26 28 29 30 32 33 34 35 36 ...
[26] 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 69
```

Nous souhaitons maintenant calculer en chaque point du design le pourcentage de malade (ou le pourcentage de 1, un malade étant codé 1). Ce pourcentage est égal à la moyenne de la variable binaire au point du design et il est donc calculé par

```
> sature <- aggregate(artere[,"chd"],by=list(artere$age),FUN=mean)
> names(sature) <- c("age","p")
```

Nous pouvons aussi calculer par le même moyen l'effectif n_t à chaque point du design t.

```
> ndesign <- aggregate(artere[,"chd"],
                 by=list(artere$age),FUN=length)
> names(ndesign) <- c("age","n")
> merge(sature,ndesign,by="age")[1:5,]
  age   p n
1  20 0.0 1
2  23 0.0 1
3  24 0.0 1
4  25 0.5 2
5  26 0.0 2
```

Nous constatons qu'au premier point du design, `age` vaut 20, il y a une seule observation et la probabilité estimée d'être malade est de 0, alors qu'au quatrième point il y a 2 répétitions ($n_4 = 2$) et la probabilité estimée d'être malade vaut 0.5. Nous pouvons ensuite représenter graphiquement ces pourcentages ainsi que les observations grâce à

```
> plot(chd ~ age, data = artere, pch = 15+chd, col = chd+1)
> lines(p ~ age, data = sature)
```

et nous retrouvons graphiquement (fig. 11.4) que le modèle saturé est une ligne brisée qui ajuste les points au mieux. Nous remarquons que ce modèle est très instable, il sera fortement affecté par de faibles perturbations des données.

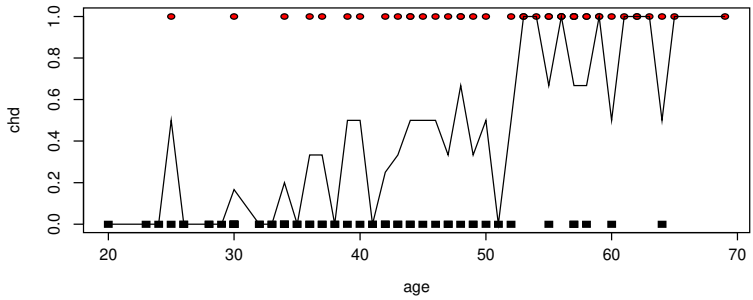


Fig. 11.4 – Modèle saturé pour l'exemple de la maladie coronarienne.

11.4.2 Tests d'adéquation de la déviance et de Pearson

Nous présentons dans cette section deux tests d'adéquation pour le modèle logistique. L'hypothèse nulle H_0 stipule que le modèle logistique considéré est le « vrai

modèle », c'est-à-dire que les observations $\tilde{y}_t$ sont des réalisations de loi Binomiale de paramètres n_t et $p_\beta(\tilde{x}_t)$ telles que

$$\mathsf{logit}(p_\beta(\tilde{x}_t)) = \beta_1 \tilde{x}_{t1} + \ldots + \beta_p \tilde{x}_{tp}, \quad t = 1, \ldots, T.$$

L'alternative H_1 est la négation de H_0. Les statistiques de test sont basées sur deux critères d'ajustement : la *déviance* et la *statistique de Pearson*.
La déviance mesure l'écart en termes de log-vraisemblance entre le modèle saturé et le modèle considéré.

Définition 11.2
La déviance du modèle logistique est définie par

$$\mathcal{D} = -2(\mathcal{L}(Y, \hat{\beta}) - \mathcal{L}_{sat}(Y, \hat{p})) \geq 0, \tag{11.21}$$

où $\mathcal{L}_{sat}$ désigne la log-vraisemblance maximisée du modèle saturé. Elle se réécrit

$$\mathcal{D} = 2 \sum_{t=1}^{T} \tilde{y}_t \log \left(\frac{\tilde{y}_t}{n_t p_{\hat{\beta}}(\tilde{x}_t)} \right) + (n_t - \tilde{y}_t) \log \left(\frac{n_t - \tilde{y}_t}{n_t - n_t p_{\hat{\beta}}(\tilde{x}_t)} \right).$$

La déviance est toujours positive (voir Propriété 11.2). C'est un indicateur qui mesure la qualité d'ajustement du modèle : plus la déviance est faible, mieux le modèle ajuste les données.
Un autre moyen de mesurer l'ajustement consiste à étudier l'écart entre les valeurs observées et les valeurs ajustées par le modèle. Pour le modèle logistique, il s'agira de confronter les observations y_i aux valeurs ajustées $p_{\hat{\beta}}(x_i)$. Plus généralement, dans le cas de données répétées, on confrontera les $\tilde{y}_t$ aux valeurs $n_t p_{\hat{\beta}}(\tilde{x}_t)$. La statistique de Pearson est définie par la somme des carrés des écarts normalisés entre $\tilde{y}_t$ à $n_t p_{\hat{\beta}}(\tilde{x}_t)$ sur tous les points de design $\tilde{x}_t$:

$$P = \sum_{t=1}^{T} \frac{(\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t))^2}{n_t p_{\hat{\beta}}(\tilde{x}_t)(1 - p_{\hat{\beta}}(\tilde{x}_t))}. \tag{11.22}$$

Les statistiques $\mathcal{D}$ et P mesurent l'adéquation du modèle logistique considéré. Si elles sont proches de 0, on peut considérer que le modèle ajuste bien les observations, si elles sont élevées l'ajustement est mauvais. Une valeur faible ne signifie cependant pas que le modèle est bon, la qualité dépend également de la complexité du modèle (c'est-à-dire du nombre de paramètres à estimer dans le modèle). Les tests d'adéquation de la déviance et de Pearson permettent de prendre en considération ce compromis « ajustement/complexité ». Sous les conditions suivantes :
— le nombre de points de design T est fixé et ne dépend pas de n ;
— les nombres de répétitions n_t sont suffisamment grands.
Les lois des statistiques $\mathcal{D}$ et P peuvent être approchées par une loi du χ^2 à $T - p$ degrés de liberté sous H_0 (on rappelle que p est le nombre de paramètres du modèle logistique). Cela signifie que si on dispose de suffisamment de répétitions en chaque point de design, on peut utiliser les statistiques $\mathcal{D}$ et P pour tester l'adéquation. Il

suffit de comparer ces statistiques au quantile d'ordre $1-\alpha$ de la loi du χ^2 à $T-p$ degrés de liberté. Si on est en présence de données individuelles, ou qu'on a peu de répétitions sur les points du design (n_t petit), ces tests ne pourront pas être utilisés. On a souvent recours au test de Hosmer et Lemeshow que nous présentons dans la partie suivante.

Test de Hosmer et Lemeshow

Ce test permet de vérifier l'adéquation du modèle logistique en présence de données individuelles (ou lorsqu'on a peu de répétitions sur les points du design). L'approche consiste à se rapprocher du cas de données répétées en créant des groupes d'observations « proches ». Une statistique de type Pearson est ensuite définie en utilisant ces groupes. On se place dans le cas de données individuelles $(x_i, y_i), i = 1, \ldots, n$. Le test s'effectue de la manière suivante (voir Hosmer & Lemeshow (2000), chapitre 5 pour plus de précisions).

1. Les probabilités $p_{\hat{\beta}}(x_i)$ sont ordonnées par ordre croissant.

2. Ces probabilités ordonnées sont ensuite séparées en K groupes de taille égale ou approximativement égales si K ne divise pas n (on prend souvent $K = 10$). On note pour k variant de 1 à K :

— m_k les effectifs du groupe k ;
— o_k le nombre de succès ($Y = 1$) observés dans le groupe k ;
— μ_k la moyenne des $p_{\hat{\beta}}(x_i)$ dans le groupe k.

La statistique de test est définie par

$$C^2 = \sum_{k=1}^{K} \frac{(o_k - m_k\mu_k)^2}{m_k\mu_k(1-\mu_k)}.$$

Le test se conduit de manière identique aux tests de déviance et de Pearson, les hypothèses sont les mêmes et la statistique C^2 suit approximativement sous H_0 une loi du χ^2 à $K-2$ degrés de liberté. Sur R la fonction **logitgof** du package **generalhoslem** permet de réaliser ce test. On testera par exemple l'adéquation du modèle logistique permettant d'expliquer la variable `chd` par les autres variables du jeu de données `SAheart` avec

```
> model <- glm(chd ~ ., data = SAheart, family = binomial)
> library(generalhoslem)
> logitgof(obs= SAheart$chd, exp = fitted(model))

        Hosmer-Lemeshow C statistic

data:  SAheart$chd, fitted(model)
X-squared = 6.6586, df = 8, p-value = 0.5739
```

La statistique de test suit une loi du χ^2 à 8 degrés de liberté (par défaut $K = 10$), on acceptera l'hypothèse nulle au niveau 5% : le modèle logistique avec toutes les variables est satisfaisant.

11.4.3 Analyse des résidus

Comme pour le modèle linéaire, les résidus permettent de mesurer l'écart entre chaque observation et sa valeur ajustée par le modèle. Les objectifs sont identiques : identifier des valeurs aberrantes, des points leviers. Nous présentons dans cette section uniquement les différents types de résidus pour le modèle logistique. Pour plus de précisions sur leur analyse, on pourra se référer au chapitre 3.

Les différents types de résidus

On reste dans le cas général de données répétées présenté dans la section précédente. Pour chaque point d'observation $\tilde{x}_t$, le modèle renvoie une estimation $p_{\hat{\beta}}(\tilde{x}_t)$ de la probabilité que Y soit égale à 1. Une première façon de mesurer l'écart entre l'observation $\tilde{y}_t$ et sa valeur ajustée par le modèle est donc de considérer $\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t)$. Ces résidus sont appelés *résidus bruts*. Ils permettent de mesurer l'ajustement du modèle sur chaque observation. Ces résidus n'ayant pas la même variance, ils sont difficiles à comparer. En effet, comme $\mathrm{V}(\tilde{y}_t) = n_t p_\beta(\tilde{x}_t)(1 - p_\beta(\tilde{x}_t))$, la variance des résidus bruts risque d'être élevée pour des valeurs de $p_\beta(\tilde{x}_t)$ proches de 1/2. Un moyen de pallier cette difficulté est de standardiser les résidus en divisant les résidus bruts par leur écart-type estimé. On obtient les *résidus de Pearson* :

$$\hat{\varepsilon}_t^P = \frac{\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t)}{\sqrt{n_t p_{\hat{\beta}}(\tilde{x}_t)(1 - p_{\hat{\beta}}(\tilde{x}_t))}}. \tag{11.23}$$

La standardisation n'est cependant pas tout à fait correcte. Elle est effectuée en prenant en compte la variance de $\tilde{y}_t$, pas celle de $\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t)$ (il ne faut pas oublier que $p_{\hat{\beta}}(\tilde{x}_t)$ est aléatoire). Pour prendre en compte cette dernière variance, on utilise le fait que la variance de $\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t)$ peut être approchée par

$$(1 - h_t)\,\mathrm{V}(\tilde{y}_t) = (1 - h_t) n_t p_\beta(\tilde{x}_t)(1 - p_\beta(\tilde{x}_t))$$

où h_t est le t^{e} élément de la diagonale de la « hat » matrice (voir Collet 2003)

$$H = \widetilde{W}_{\hat{\beta}}^{1/2} \widetilde{X} (\widetilde{X}' \widetilde{W}_{\hat{\beta}} \widetilde{X})^{-1} \widetilde{X}' \widetilde{W}_{\hat{\beta}}^{1/2},$$

où $\widetilde{X}$ est la matrice $T \times p$ contenant les points de design et $\widetilde{W}_{\hat{\beta}}$ la matrice diagonale $T \times T$ dont le t^{e} élément de la diagonale vaut $n_t p_{\hat{\beta}}(x_t)(1 - p_{\hat{\beta}}(x_t))$. On peut alors définir la version standardisée des résidus de Pearson par

$$\hat{\varepsilon}_t^{PS} = \frac{\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t)}{\sqrt{n_t p_{\hat{\beta}}(\tilde{x}_t)(1 - p_{\hat{\beta}}(\tilde{x}_t))(1 - h_t)}}, \quad t = 1, \ldots, T.$$

On remarque que la statistique de test de Pearson (11.22) s'écrit comme la somme des carrés des résidus de Pearson (11.23). Cela donne une nouvelle interprétation à cette statistique : elle est définie comme la somme des écarts (au sens des résidus de Pearson) entre chaque observation et sa valeur ajustée. En procédant de façon identique pour la déviance, on obtient les *résidus de déviance*

$$\hat{\varepsilon}_t^D = \text{signe}(\tilde{y}_t - n_t p_{\hat{\beta}}(\tilde{x}_t))\sqrt{2\left[\tilde{y}_t \log\left(\frac{\tilde{y}_t}{n_t p_{\hat{\beta}}(\tilde{x}_t)}\right) + (n_t - \tilde{y}_t)\log\left(\frac{n_t - \tilde{y}_t}{n_t - n_t p_{\hat{\beta}}(\tilde{x}_t)}\right)\right]}.$$

Pour tenir compte de la variabilité, ces résidus peuvent être standardisés :

$$\hat{\varepsilon}_t^{DS} = \frac{\hat{\varepsilon}_t^D}{\sqrt{1-h_t}}.$$

Les résidus de Pearson et de déviance sont les plus utilisés et sont généralement renvoyés par les logiciels statistiques.

Remarque

— Les résidus de déviance sont le plus souvent privilégiés. En effet, lorsque les nombres de répétitions n_t sont grands, ces résidus suivent approximativement une loi $\mathcal{N}(0,1)$. Leur analyse est ainsi similaire à celle du modèle de régression linéaire. En présence de données individuelles, nous n'avons pas d'information précise sur la loi de ces résidus. On sait juste qu'ils sont approximativement d'espérance nulle et de variance 1.

— Il existe d'autres types de résidus pour le modèle de régression logistique. On pourra par exemple regarder les résidus partiels qui s'interprètent de la même façon que dans le modèle linéaire (voir section 3.4.3) et permettent d'identifier des effets non linéaires dans le modèle. Sur R on obtiendra ces résidus à l'aide de la commande :

```
> residuals(model,type="partial")
```

Analyse des résidus

Les analyses sont essentiellement graphiques et consistent à tracer des « index plot » avec en abscisse le numéro de l'observation et en ordonnée le résidu ou encore à représenter le résidu $\hat{\varepsilon}_t$ en fonction de la combinaison linéaire $\tilde{x}_t'\hat{\beta}$. En présence de données individuelles, on observera fréquemment une structure sur ce type de représentation. C'est le cas sur l'exemple suivant : commençons par estimer les paramètres du modèle complet avec toutes les variables explicatives du jeu de données `SAheart` :

```
> model <- glm(chd ~ ., data = SAheart, family = binomial)
```

En utilisant ce modèle, nous pouvons ajuster les observations sur l'échelle linéaire $X\hat{\beta}$ grâce à l'ordre suivant :

```
> prev_lin <- predict(model)
```

Puis nous obtenons les différents types de résidus par :

```
> res_P <- residuals(model, type = "pearson") #Pearson
> res_PS <- rstandard(model, type = "pearson") #Pearson standard
> res_D <- residuals(model, type = "deviance")  #Deviance
> res_DS <- rstandard(model, type = "deviance") #Deviance standard
```

Enfin la représentation des résidus (voir figure 11.5) est obtenue par :

```
> par(mfrow=c(2, 2), pch=20)
> plot(res_PS, xlab="index", ylab="Pearson Standard")
> plot(prev_lin, res_PS, xlab="Prevision lineaire",
                                  ylab="Pearson Standard")
> plot(res_DS, xlab="index" ,ylab="Deviance Standard")
> plot(prev_lin, res_DS, xlab="Prevision lineaire",
                                 ylab="Deviance Standard")
```

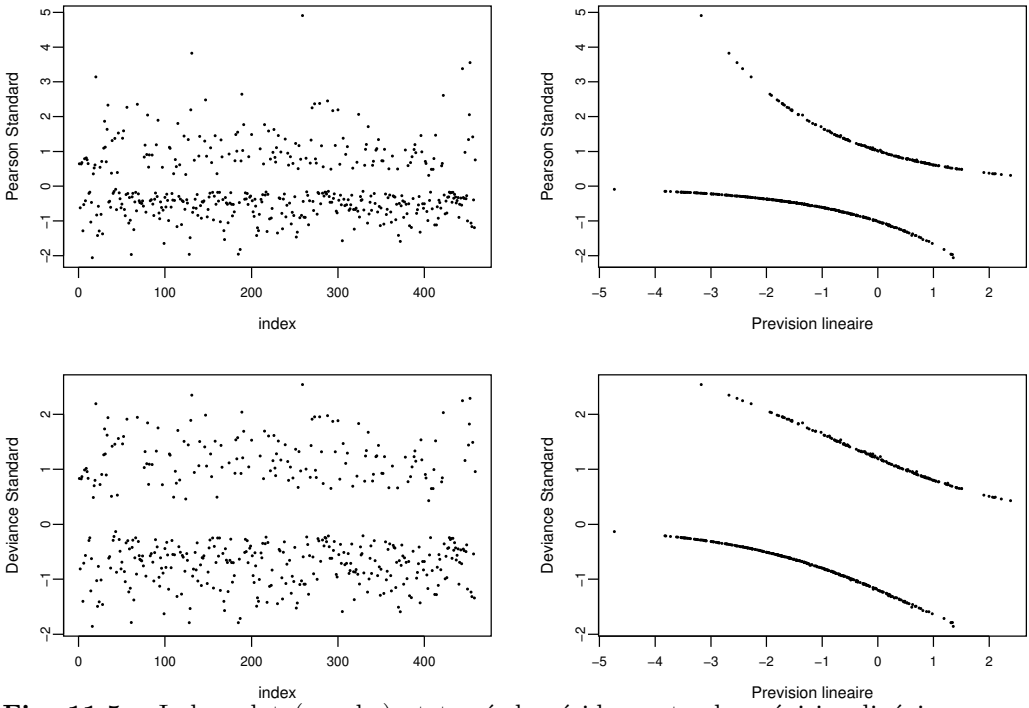


Fig. 11.5 – Index plot (gauche) et tracé du résidu contre la prévision linéaire (droite) pour les résidus de Pearson (haut) et de déviance (bas) standardisés.

Nous observons sur les graphiques de droite (figure 11.5) une structure particulière. Ce type de structure apparaît fréquemment en présence de données individuelles et rend peu utile ce type de représentation. Les graphiques de gauche ou « index plot » sont plus aisés à lire et sont donc conseillés, notamment celui en bas à gauche (résidus de déviance standardisés). Ici aucun point ne semble très loin de l'intervalle (-2,2) et le modèle ne présente donc pas de point visiblement aberrant (avec de fortes valeurs de résidus). Les résidus de Pearson ont des variations plus exagérées.

Résidus partiels

Nous reprenons ici les résidus partiels présentés dans la section 3.4.3 (p. 65). Ces résidus servent à savoir, pour chaque variable explicative, si elle intervient de manière linéaire (comme prévu par le modèle) ou si il faut ajouter une fonction non-linéaire de cette variable explicative. Rappelons la définition des résidus partiels (ici la i^{e} coordonnée) dans le cadre du modèle linéaire pour la variable X_j :

$$[\hat{\varepsilon}_P^j]_i = (y_i - \hat{y}_i) + \hat{\beta}_j X_{ij},$$

où $\hat{y}_i = x_i'\hat{\beta}$ est la prévision (ou ajustement) de la valeur y_i mais aussi de la valeur moyenne $\mathbb{E}(y_i) = x_i'\beta$. Nous avons donc

— un premier terme qui est l'écart entre une valeur observée et l'ajustement de la valeur moyenne (terme $\Delta_y = y_i - \hat{y}_i$) ;

— additionné à un second terme qui l'effet de la variable j et qui vaut $\hat{\beta}_j X_j$.

Pour adapter cette définition à notre cadre rappelons la définition de notre modèle :

$$\log\left(\frac{p_\beta(x_i)}{1 - p_\beta(x_i)}\right) = \beta_1 x_{i1} + \ldots + \beta_p x_{ip} = x_i'\beta.$$

— Dans le premier terme, la prévision de la valeur moyenne en i, notée $p_\beta(x_i)$, est ici $p_{\hat{\beta}}(x_i)$. L'écart entre une valeur observée et l'ajustement de la valeur moyenne est simplement $\Delta_y = y_i - p_{\hat{\beta}}(x_i)$;

— Le second terme est toujours l'effet de la variable j (au point i) valant $\hat{\beta}_j X_{ij}$;

— Mais pour additionner ces deux termes, il faut qu'ils soient sur la même échelle, ce qui n'est plus le cas ici. En effet, l'équation du modèle montre que pour passer de l'échelle des observations y_i ou de leur moyenne $p_\beta(x_i)$ à l'échelle des $\beta_j x_{ij}$ nous utilisons la fonction logit car $\mathsf{logit}(p_\beta(x_i)) = \log\left(\frac{p_\beta(x_i)}{1-p_\beta(x_i)}\right)$.
Cette fonction à valeur réelle $p \mapsto x = \mathsf{logit}(p)$ est différentiable à l'ordre 1 et nous avons donc

$$\Delta_x = \mathsf{logit}'(p_{\hat{\beta}}(x_i))\Delta_y + o(\Delta_y).$$

En partant de l'écart sur l'échelle des observations Δ_y il faut le multiplier par $\mathsf{logit}'(p_{\hat{\beta}}(x_i))$ pour obtenir un écart sur l'échelle (des effets) des variables. Le calcul de la dérivée $\mathsf{logit}'(p_{\hat{\beta}}(x_i))$ vaut ici $\frac{1}{p_{\hat{\beta}}(x_i)(1-p_{\hat{\beta}}(x_i))}$.

En conclusion, nous avons la définition suivante.

Définition 11.1 (Résidus partiels du modèle logistique)
Pour le modèle logistique, les résidus partiels pour la variable X_j sont par définition un vecteur de $\mathbb{R}^n$ dont la i^e coordonnée vaut :

$$[\hat{\varepsilon}_P^j]_i = \frac{(y_i - p_{\hat{\beta}}(x_i))}{p_{\hat{\beta}}(x_i)(1 - p_{\hat{\beta}}(x_i))} + \hat{\beta}_j x_{ij}.$$

Ces résidus s'analysent comme dans le modèle de régression. Si le le graphique des n points $\left\{(X_{ij}, [\hat{\varepsilon}_P^j]_i)\right\}_{i=1}^n$ montre une tendance linéaire alors la variable X_j intervient bien de manière linéaire, le modèle est correct pour la variable j. Si par contre la tendance sur le graphique est non linéaire selon une fonction $f(.)$, il sera bon de remplacer X_j par $f(X_j)$. Pour un exemple concret d'utilisation, voir la section 14.5.3 (p. 344). Dans le modèle logistique, il existe aussi les résidus partiels augmentés qui dans certaines situations (Mallows, 1986), peuvent être meilleurs que les résidus partiels

Définition 11.2 (Résidus partiels augmentés du modèle logistique)
Pour le modèle logistique, les résidus partiels augmentés pour la variable X_j sont par définition un vecteur de $\mathbb{R}^n$ dont la i^e coordonnée vaut :

$$[\hat{\varepsilon}_{AP}^j]_i \quad = \quad \frac{(y_i - p_{\hat{\alpha}}(x_i))}{p_{\hat{\alpha}}(x_i)(1 - p_{\hat{\alpha}}(x_i))} + \hat{\alpha}_j X_{ij} + \hat{\alpha}_{p+1} X_{ij}^2,$$

où $\hat{\alpha}$ est l'estimation du paramètre dans le modèle logistique augmenté d'un terme quadratique pour la j^e variable : $\mathsf{logit}(p_\alpha(x_i)) = \alpha_1 X_{i1} + \ldots + \alpha_p X_{ip} + \alpha_{p+1} X_{ij}^2$.

11.5 Choix de variables

Les motivations concernant le choix des variables à inclure dans le modèle logistique sont quasiment identiques à celles du modèle linéaire : oublier des variables pertinentes risque d'introduire du biais dans les estimateurs, inclure des variables inutiles va augmenter la variance des estimateurs... De plus, du point de vue de l'utilisateur, il est souvent important de déterminer parmi $\{X_1, \ldots, X_p\}$ le « meilleur » sous-groupe de variables permettant d'expliquer Y. Un tel choix s'effectue à l'aide de critères de choix de modèles. Nous présentons dans cette section les critères usuels qui permettent de sélectionner des variables dans un modèle logistique.

11.5.1 Tests entre modèles emboîtés

A l'image de ce qui a été fait dans la section 5.5.2, nous souhaitons comparer un modèle restreint avec p_0 paramètres au modèle global (avec p paramètres). Soit $p_0 < p$, on souhaite comparer le modèle $\mathcal{M}_0$

$$\mathsf{logit}(p_\gamma(x)) = \gamma_1 x_1 + \ldots + \gamma_{p_0} x_{p_0}$$

avec le modèle global $\mathcal{M}_1$

$$\mathsf{logit}(p_\beta(x)) = \beta_1 x_1 + \ldots + \beta_p x_p.$$

Une telle comparaison peut s'effectuer à l'aide d'un test d'hypothèses. En effet, plaçons-nous dans $\mathcal{M}_1$ et considérons les hypothèses

$$\mathrm{H}_0 : \quad \beta_{p_0+1} = \ldots = \beta_p = 0 \quad \text{contre} \quad \mathrm{H}_1 : \ \exists j \in \{p_0+1, \ldots, p\} : \beta_j \neq 0.$$

On remarque que accepter H_0 signifie que l'on peut privilégier $\mathcal{M}_0$ au détriment de $\mathcal{M}_1$. Ce test peut se réaliser à l'aide des procédures (Wald et rapport de vraisemblance) présentées dans la section 11.2.3. A titre d'exemple, nous proposons de comparer ici deux modèles emboîtés sur les données `SAheart` à l'aide du test du rapport de vraisemblance :

```
> model0 <- glm(chd ~ sbp + ldl, data = SAheart, family=binomial)
> model1 <- glm(chd ~ sbp + ldl + famhist + alcohol, data=SAheart,
+                  family = binomial)
> anova(model0, model1, test = "LRT")
Analysis of Deviance Table

Model 1: chd ~ sbp + ldl
Model 2: chd ~ sbp + ldl + famhist + alcohol
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       456     548.18
2       454     522.64  2   25.545 2.838e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Le test rejetant l'hypothèse nulle, on choisira le modèle à 4 variables.

11.5.2 Procédures automatiques

L'approche précédente permet uniquement de choisir un modèle parmi deux modèles emboîtés. Elle ne permet pas de sélectionner automatiquement un sous-groupe de variables explicatives. En présence de p variables explicatives $X_1, \ldots, X_p$, 2^p modèles peuvent être envisagés (sans prendre en compte les interactions). L'approche la plus naturelle pour sélectionner un modèle parmi ces 2^p revient à construire tous ces modèles et à choisir celui qui optimise un critère de choix de modèle donné. Cet algorithme est appelé *Best subset selection* (voir algorithme 5).

A nombre de variables k fixé, l'algorithme utilise la vraisemblance pour choisir le meilleur modèle à k variables (étape 1). La vraisemblance ne peut en revanche pas être utilisée pour choisir un modèle parmi les $\mathcal{M}_j, j = 0, \ldots, d$. En effet la vraisemblance augmente avec le nombre de paramètres et il est facile de voir que $\mathcal{M}_d$ est le modèle qui aura la plus grande vraisemblance. C'est pourquoi l'étape 2 utilise un critère de choix de modèles pour choisir parmi les $\mathcal{M}_j, 0, \ldots, d$.

Algorithme 5 Best subset selection

1. Pour $k = 0, \ldots, d$:

(a) Construire les $\binom{d}{k}$ modèles logistiques à k variables ;

(b) Choisir parmi ces modèles celui qui a la plus grande vraisemblance. On note $\mathcal{M}_k$ le modèle sélectionné.

2. Choisir, parmi $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_d$, le meilleur modèle au sens d'un critère donné.

Parmi les critères les plus utilisés, on retrouve, comme pour le modèle linéaire, l'AIC et le BIC. Ces critères pénalisent l'opposé de la log-vraisemblance d'un modèle $\mathcal{M}$ par son nombre de paramètres $|\mathcal{M}|$:

$$\mathrm{AIC}(\mathcal{M}) = -2\mathcal{L}_{\mathcal{M}}(Y, \hat{\beta}) + 2|\mathcal{M}| \quad \text{et} \quad \mathrm{BIC}(\mathcal{M}) = -2\mathcal{L}_{\mathcal{M}}(Y, \hat{\beta}) + |\mathcal{M}| \log(n), \tag{11.24}$$

où $\mathcal{L}_{\mathcal{M}}(Y, \hat{\beta})$ désigne la log-vraisemblance maximisée du modèle logistique $\mathcal{M}$. Ces critères sont basés sur deux parties :

— la composante $-2\mathcal{L}_{\mathcal{M}}(Y, \hat{\beta})$ mesure l'ajustement du modèle aux données (valeurs faibles pour de bons ajustements) ;

— les composantes $2|\mathcal{M}|$ pour l'AIC et $|\mathcal{M}| \log(n)$ pour le BIC mesurent la complexité du modèle.

Ces critères (à minimiser) sélectionneront donc des modèles qui réalisent un bon compromis entre qualité d'ajustement et complexité.

Remarque
Puisque $\log(n)$ est généralement plus grand que 2 (dès que $n \geq 8$), la pénalité BIC est plus grande que la pénalité AIC. Par conséquent, le critère BIC aura tendance à choisir des modèles plus parcimonieux que le critère AIC.

Exemple 11.7
Sur R, le package **bestglm** permet de choisir un modèle avec l'algorithme 5. On l'utilise ici pour sélectionner des variables sur les données (complètes) `SAheart`.

```
> library(bestglm)
> data(SAheart)
> mod_sel <- bestglm(SAheart, family = binomial)
Morgan-Tatar search since family is non-gaussian.
> mod_sel$BestModels
    sbp tobacco   ldl adiposity famhist typea obesity alcohol  age   Crit
1 FALSE    TRUE  TRUE     FALSE    TRUE  TRUE   FALSE   FALSE TRUE  506.4
2 FALSE    TRUE FALSE     FALSE    TRUE  TRUE   FALSE   FALSE TRUE  509.3
3 FALSE    TRUE  TRUE     FALSE    TRUE FALSE   FALSE   FALSE TRUE  510.0
4 FALSE   FALSE  TRUE     FALSE    TRUE  TRUE   FALSE   FALSE TRUE  510.6
5 FALSE    TRUE  TRUE     FALSE    TRUE  TRUE    TRUE   FALSE TRUE  510.8
```

On peut lire les 5 meilleurs modèles sélectionnés au sens du BIC (c'est le critère de choix de modèle par défaut dans la fonction **bestglm**). Le modèle sélectionné sera donc le modèle logistique ayant pour variables explicatives `tobacco`, `ldl`, `famhist`, `typea` et `age`. Si on veut changer le critère de choix de modèles, il suffit de modifier l'argument `IC` de la fonction **bestglm**. Pour l'AIC, on utilisera

```
> mod_sel1 <- bestglm(SAheart, family = binomial, IC = "AIC")
Morgan-Tatar search since family is non-gaussian.
> mod_sel1$BestModels
    sbp tobacco  ldl adiposity famhist typea obesity alcohol  age    Crit
1 FALSE    TRUE TRUE     FALSE    TRUE  TRUE   FALSE   FALSE TRUE  485.69
2 FALSE    TRUE TRUE     FALSE    TRUE  TRUE    TRUE   FALSE TRUE  485.98
3  TRUE    TRUE TRUE     FALSE    TRUE  TRUE    TRUE   FALSE TRUE  486.55
4  TRUE    TRUE TRUE     FALSE    TRUE  TRUE   FALSE   FALSE TRUE  486.65
5 FALSE    TRUE TRUE      TRUE    TRUE  TRUE   FALSE   FALSE TRUE  487.44
```

Sur cet exemple, l'AIC sélectionne le même modèle que le BIC.

Remarque
La procédure exhaustive présentée dans l'algorithme 5 nécessite le calcul de 2^p modèles logistiques. Elle peut donc devenir très coûteuse en temps de calcul lorsque le nombre de variables p est grand (au-delà de $p = 30$ généralement). Dans ce cas, les procédures pas à pas (ascendantes, descendantes et progressives) sont alors nécessaires. Ces méthodes sont identiques au cas de la régression linéaire et nous renvoyons le lecteur à la section 7.4.2 pour plus de précisions. Sur R, on peut utiliser la fonction **step** ou l'argument `method=backward` (ou `method=forward`) dans la fonction **bestglm** pour lancer ces procédures pas à pas.

11.6 Exercices

Exercice 11.1 (Questions de cours)

1) Nous souhaitons effectuer une régression logistique, on utilise la fonction **glm** en précisant
   A. family=binomial,
   B. en ne précisant rien,
   C. family=poisson.
2) Lors d'une régression logistique (hors modèle saturé ou constant), les estimateurs sont obtenus en utilisant un algorithme itératif :
   A. oui toujours,
   B. non jamais,
   C. seulement si les variables explicatives sont qualitatives.
3) Un estimateur de la variance de $\hat{\beta}$ de β dans le cas de la régression logistique vaut :
   A. $\hat{\sigma}^2(X'X)^{-1}$,
   B. $(X'W_{\hat{\beta}}X)^{-1}$,
   C. $\hat{\sigma}^2(W_{\hat{\beta}})^{-1}$.
4) Pour estimer les paramètres d'une régression logistique, on
   A. maximise la vraisemblance,

B. minimise le nombre de faux positifs,
C. minimise les moindres carrés.

5) Les probabilités p_t du modèle « saturé » en chaque unique point du design $\tilde{x}_t$ (avec $\tilde{x}_t \neq \tilde{x}_l$ pour $t \neq l$) sont estimées par
A. la moyenne $\frac{1}{n_t}\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$,
B. la somme $\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$,
C. l'inverse du logit appliqué à la somme $\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$.

6) Le modèle de régression logistique
A. impose qu'en chaque unique point du design $\tilde{x}_t$ (avec $\tilde{x}_t \neq \tilde{x}_l$ pour $t \neq l$) la somme des $y_i \mathbb{1}_{x_t}(x_i)$ suive une loi binomiale,
B. impose que les y_i suivent une loi de Poisson.

7) Les modèles de régression logistique
A. imposent qu'en chaque unique point du design la variance est constante (égale à σ^2),
B. imposent qu'en chaque unique point du design la variance vaut $n_t p_t(1-p_t)$,
C. imposent qu'en chaque unique point du design la variance vaut $n_t p_t$.

8) Les intervalles de confiances en régression logistique (de niveau $1-\alpha$) pour les coordonnées de β
A. sont approximativement de niveau $1-\alpha$,
B. sont précisément de niveau $1-\alpha$.

Exercice 11.2 (Interprétation des coefficients)
On considère X une variable aléatoire qualitative qui suit une loi uniforme de support $\{A, B, C\}$ et Y une variable dichotomique telle que :

$$Y|X = A \sim \mathrm{Ber}(0.95), \quad Y|X = B \sim \mathrm{Ber}(0.95), \quad Y|X = C \sim \mathrm{Ber}(0.05).$$

1) Simuler un échantillon $(X_i, Y_i), i = 1, \ldots, n$ de taille $n = 100$ tirés selon la loi de (X, Y).
2) Construire le modèle 1 :

$$\text{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x)$$

muni de la contrainte $\beta_1 = 0$ (on pourra utiliser la fonction **glm**).
a) Donner les estimations des paramètres.
b) Faire un **summary** du modèle et analyser les sorties (on précisera notamment les tests effectués).

3) Construire le modèle 2 :

$$\text{logit}(p_\beta(x)) = \beta_0 + \beta_1 \mathbf{1}_A(x) + \beta_2 \mathbf{1}_B(x) + \beta_3 \mathbf{1}_C(x)$$

muni de la contrainte $\beta_3 = 0$ (on pourra utiliser la fonction **glm**).
a) Donner les estimations des paramètres.
b) Faire un `summary` du modèle et analyser les sorties (on précisera notamment les tests effectués).

4) Comparer les probabilités critiques du test $H_0 : \beta_2 = 0$ contre $H_1 : \beta_2 \neq 0$ pour les deux modèles construits. Interpréter.
5) Proposer et mettre en œuvre un test statistique permettant de répondre à la question : Peut-on dire que X à un effet sur Y ?

Exercice 11.3 (Séparabilité)
1) Générer un échantillon $(x_i, y_i), i = 1, \ldots, 100$ tel que

— pour $i = 1, \ldots, 50$, x_i est le réalisation d'une loi uniforme sur $[-1, 0]$ et $y_i = 0$;
— pour $i = 51, \ldots, 100$, x_i est le réalisation d'une loi uniforme sur $[0, 1]$ et $y_i = 1$.

2) On considère le modèle logistique permettant d'expliquer Y par X sans la constante :

$$\text{logit}(p_\beta(x)) = \beta x.$$

Représenter sur un graphe la log-vraisemblance de ce modèle en fonction de β. On pourra faire varier β entre 0 et 100.

3) Estimer le paramètre du modèle logistique à l'aide de la fonction **glm**. Que remarquez-vous ?

4) On considère le même échantillon en remplaçant la première valeur de Y par 1 ($y_1 = 1$). Refaire les questions 2 et 3 avec ces données.

Le problème mis en avant ici est celui de la séparabilité des données. Dans le premier échantillon, les deux groupes sont en effet parfaitement séparés puisque Y vaut 1 lorsque X est positif et 0 lorsque X est négatif. Dans ce cas, la vraisemblance tend vers 1 lorsque β tend vers $+\infty$ et il n'existe donc pas de solution finie au problème de maximisation de la vraisemblance. Cette notion de séparabilité des données est présentée en détails dans l'article de Albert & Anderson (1984).

Exercice 11.4 (Matrice hessienne)
Montrer que la matrice hessienne de la log-vraisemblance (11.7) est définie négative si X est de plein rang et en déduire le proposition 11.1.

Exercice 11.5 (Modèles avec R)
On souhaite expliquer des pannes de machines. Pour ce faire, on dispose :

— d'une variable `etat` qui vaut 1 si la machine est en panne, 0 sinon ;
— d'une variable `age` qui correspond à l'âge de la machine ;
— d'une variable `marque` qui représente la marque de la machine (A, B ou C).

Voici les 5 premiers individus du jeu de données :

```
> head(panne)
   etat age marque
1     0   4      A
2     1   2      C
3     0   3      C
4     0   9      B
5     1   7      B
```

1) On effectue :

```
> model <- glm(etat ~ ., data = panne, family = binomial)
```

```
> model

Call:  glm(formula = etat ~ ., family = binomial, data = panne)

Coefficients:
(Intercept)          age      marqueB      marqueC
    0.47808      0.01388     -0.41941     -1.45608

Degrees of Freedom: 32 Total (i.e. Null);  29 Residual
Null Deviance:            45.72
Residual Deviance: 43.5         AIC: 51.5
```

Ecrire le modèle ajusté et donner les estimations des paramètres du modèle.
2) On effectue :

```
> library(car)
> Anova(model, type = 3, test.statistic = "Wald")
Analysis of Deviance Table (Type III tests)

Response: etat
            Df  Chisq Pr(>Chisq)
(Intercept)  1 0.3294     0.5660
age          ? 0.0218     0.8826
marque       ? 1.9307     0.3809
```

Ecrire le test correspondant à la ligne `marque` du tableau ci-dessus (on donnera les hypothèses, la statistique de test, sa loi sous H_0 et la conclusion du test. On complètera également le tableau (en remplaçant les « ? »)).
3) Même question que précédemment pour la commande :

```
> Anova(model, type = 3, test.statistic = "LR")
Analysis of Deviance Table (Type III tests)

Response: etat
       LR Chisq Df Pr(>Chisq)
age     0.02189  ?     0.8824
marque  2.09562  ?     0.3507
```

4) On souhaite expliquer l'état de la machine par la marque **uniquement**.
   a) Ecrire le modèle logistique permettant de réaliser cela (on prendra comme modalité de référence la modalité `C` de la variable `marque`).
   b) On ajuste le modèle sur R :

```
> glm(etat ~ marque, data = panne, family = binomial)

Coefficients:
(Intercept)      marqueB      marqueC
     0.56        -0.43        -1.48
```

Donner les estimations des paramètres du modèle écrit à la question précédente.

c) Ecrire le modèle utilisant comme contrainte la nullité de la somme des coefficients des modalités de `marque`. Donner les estimations des paramètres de ce modèle.

5) Donner la définition de l'interaction entre les variables X_1 et X_2.

6) Soit $x = (x_1, x_2)$ où x_1 représente l'âge d'une machine et x_2 sa marque. On considère le modèle

$$\text{logit} p_\beta(x) = \begin{cases} \alpha_0 + \alpha_1 x_1 & \text{si } x_2 = A \\ \beta_0 + \beta_1 x_1 & \text{si } x_2 = B \\ \gamma_0 + \gamma_1 x_1 & \text{si } x_2 = C \end{cases}$$

On lance sur R :

```
> glm(etat ~ marque + age + age:marque, data = panne, family = binomial)

Coefficients:
(Intercept)    marqueB    marqueC     age  marqueB:age  marqueC:age
    0.24       0.20       -2.43       0.06       -0.11         0.22
```

Calculer les estimations des paramètres $\alpha_j, \beta_j, \gamma_j, j = 0, 1$.

Exercice 11.6 (Interprétation)

Le fichier `logit_ex6.csv` contient $n = 500$ observations d'une variable binaire Y et de deux variables continues X_1 et X_2. On cherche à expliquer Y par X_1 et X_2 à l'aide d'un modèle logistique.

1) Construire sur R le modèle logistique avec les deux variables explicatives X_1 et X_2. Interpréter les tests de nullité des coefficients du modèle.

2) Même question pour le modèle logistique possédant uniquement X_1 comme variable explicative.

3) Interpréter.

Exercice 11.7 (Tests "à la main")

On dispose de $n = 5$ observations $(x_1, y_1), \ldots, (x_n, y_n)$ telles que $x_i \in \mathbb{R}$ et $y_i \in \{0, 1\}$. Les x_i correspondent à des observations d'une variable X supposée déterministe et les y_i correspondent à des observations d'une variable Y. On cherche à expliquer Y par X. Les données sont dans le tableau suivant.

x_i	0.47	-0.55	-0.01	1.07	-0.71
y_i	1	0	0	1	1

1) Ecrire le modèle de régression logistique permettant d'expliquer Y par X (on inclura la constante dans le modèle).

2) Les estimations $p_{\hat\beta}(x_i)$ des probabilités $\Pr(Y_i = 1)$ par ce modèle sont

$$p_{\hat\beta}(x_1) = 0.76,\ p_{\hat\beta}(x_2) = 0.40,\ p_{\hat\beta}(x_3) = 0.60,\ p_{\hat\beta}(x_4) = 0.89,\ p_{\hat\beta}(x_5) = 0.35.$$

Calculer la log-vraisemblance maximisée $\mathcal{L}_n(\hat\beta)$ du modèle.

3) On trouve ci-dessous un extrait du tableau des coefficients de ce modèle ajusté sur R.

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.4383     ???       ???     ???
X             1.5063     ???       ???     ???
```

a) Compléter les colonnes `Std. Error` et `z value`.
b) Proposer deux procédures de test permettant de tester la nullité du coefficient associé à la variable explicative X. On pourra prendre un niveau de test de $\alpha = 5\%$.
c) Effectuer les deux tests.

Exercice 11.8 (Vraisemblance du modèle saturé)
Démontrer la proposition 11.2.

Exercice 11.9 († Intervalle de confiance profilé)
Pour un modèle de régression logistique à une variable explicative X (logit$(p_\beta(x)) = \beta_1 + \beta_2 x$), un intervalle de confiance pour β_2 (par exemple) basé sur la vraisemblance $\mathcal{L}(Y, \beta_1, \beta_2)$ utilise la vraisemblance « profilée ». La vraisemblance « profilée » pour le paramètre d'intérêt β_2 est une fonction de β_2 qui vaut $l(\beta_2) = \max_{\beta_1 \in} \mathcal{L}(Y, \beta_1, \beta_2)$. Un intervalle de confiance de niveau $1 - \alpha$ est alors

$$(\beta_2^-, \beta_2^+) = \{\beta_2^* \in \mathbb{R} \mid 2(\mathcal{L}(Y, \hat{\beta}_1, \hat{\beta}_2) - l(\beta_2^*)) \leq q_1(1-\alpha)\},$$

où $q_1(1-\alpha)$ est le quantile de niveau $1-\alpha$ d'une loi $\chi^2(1)$. Notons pour la suite de cet exercice $P(\beta_2^*) = 2(\mathcal{L}(Y, \hat{\beta}_1, \hat{\beta}_2) - l(\beta_2^*))$.
Afin de trouver β_2^- et β_2^+ les deux bornes de cet intervalle, nous calculons sur une grille de valeurs de β_2^* « bien choisies », les valeurs de profil $P(\beta_2^*)$ correspondantes. Comme il est peu probable de tomber exactement sur deux valeurs de la grille qui forment les bornes de l'intervalle recherché, une étape d'interpolation supplémentaire sera nécessaire.

1) Ajuster le modèle pour le jeu de données `artere.txt`. Extraire les coefficients estimés (fonction **coef**) et les assigner dans un vecteur nommé `B0`. Le modèle logistique est donc le suivant (1) : les $\{y_i\}_{i=1}^n$ (`chd`) sont iid et suivent une Bernoulli de paramètre $p_\beta(x_i)$ avec logit$(p_\beta(x)) = \beta_1 + \beta_2 x$ (où X dénote la variable age). Affecter dans `OriginalDeviance` la déviance du modèle (composante `deviance` du modèle obtenu).
2) Le niveau $1-\alpha$ de l'intervalle de confiance sera choisi égal à 95% et donc $\alpha = 0.05$.
3) Construisons une grille d'évaluation autour de $\hat{\beta}_2$ de la forme suivante : $\ldots, \hat{\beta}_2 - 2\delta, \hat{\beta}_2 - \delta, \hat{\beta}_2, \hat{\beta}_2 + \delta, \hat{\beta}_2 + 2\delta, \ldots$.
Pour cela nous souhaitons faire assez peu de calculs et faire une grille d'une longueur de 10-20 valeurs. On sait par construction que $P(\hat{\beta}_2) = 0$ et plus nous éloignons de $\hat{\beta}_2$ vers le bas ou vers le haut, plus $P(\beta_2^*)$ augmente vers $q_1(1-\alpha)$.
La grille va être construite comme suit : le pas δ sera égal à $\sqrt{q_1(1-\alpha/4)} * \hat{\sigma}_{\hat{\beta}_2}/5$.
Affecter le résumé du modèle (fonction `summary`) dans un objet provisoire (de type liste). De cette liste, extraire les écarts-types estimés des coefficients (composante `coefficients`) que l'on affectera dans l'objet `stderr`. Calculer δ en utilisant l'écart-type estimé et la fonction `qchisq`. Construire un vecteur `grille` avec des valeurs variant de $\pm 10.\delta$ autour de $\hat{\beta}_2$.
4) Pour chaque coordonnée de `grille` nous allons calculer la fonction de vraisemblance profilée. Pour cela nous utiliserons la fonction `glm` et l'argument `offset`. Rappelons que la valeur de β_2^* est fixée à une valeur de grille, par exemple $\hat{\beta}_2 + 3\delta$. Pour cette valeur nous devons estimer par maximum de vraisemblance les paramètres du modèle restant (il n'en reste qu'un : β_1). Pour cela nous allons utiliser la notion d'offset. L'offset est, dans un modèle GLM, un vecteur K de coordonnées $K_1, \ldots, K_n$ de valeurs fixées qui se rajoute à la partie linéaire. Pour un modèle logistique à une variable X et un offset K, nous avons donc pour la i^e observation

$$\text{logit}(p_\beta(x_i, K_i)) = K_i + \beta_1 + \beta_2 x_i.$$

L'ajustement de ce modèle conduira à l'estimation de 2 paramètres β_1 et β_2 ce qui n'est pas ce que nous souhaitons ! Par contre, le modèle avec uniquement la constante (*intercept* en anglais) et un offset $K_i = (\hat{\beta}_2 + 3\delta)x_i$ va donner

$$\text{logit}(p_{\beta_1}(K_i)) = K_i + \beta_1 = \beta_1 + (\hat{\beta}_2 + 3\delta)x_i$$

et conduira à estimer par maximum de vraisemblance uniquement β_1 (rappelons que l'offset est fixe) et nous donnera donc la vraisemblance profilée au point voulu : $l(\hat{\beta}_2 + 3\delta)$.
En se rappelant la définition de la déviance, montrer que $P(\hat{\beta}_2 + 3\delta)$ est simplement la différence de la déviance du modèle avec offset et la déviance du modèle (1) de départ.
5) Pour toutes les valeurs de k utilisées pour `grille`, calculer les valeurs de $P(\hat{\beta}_2 + k\delta)$ et stocker le résultat dans `profil2`.
6) Calculer la racine carrée de `profil2` et lui donner un signe positif quand $k > 0$ et négatif quand $k < 0$. Le résultat sera nommé `profil` (il s'agit bien de $\sqrt{P(.)}$ (signée) pour différentes valeurs de la grille et non pas de $P(.)$ mais c'est sur cette échelle que travaille R).
7) En utilisant `grille`, `profil` et la fonction d'interpolation `spline`, trouver les 2 valeurs qui permettent d'obtenir les valeurs de `profil` égales à $\pm\sqrt{q_1(1-\alpha)}$.
8) Comparer ces valeurs à celles obtenues avec la fonction `confint` (le logiciel R n'utilise qu'une partie des valeurs de `grille`, ce qui explique la légère différence numérique).

Chapitre 12

Régression de Poisson

Les modèles linéaire et logistique étudiés dans les chapitres précédents présentent de nombreuses similitudes. On peut utiliser un protocole identique pour les définir qui consiste à d'abord choisir une loi pour y_i puis à trouver une fonction pertinente qui permet de faire le lien entre l'espérance de y_i et une combinaison linéaire des variables explicatives x_i.
En effet pour le modèle linéaire gaussien, on suppose que y_i suit une loi normale et on lie l'espérance de y_i aux variables x_i selon

$$g(\mathbb{E}[y_i]) = \mathbb{E}[y_i] = x_i'\beta,$$

la fonction utilisée pour ce lien est donc la fonction identité.
Dans le cas du modèle logistique, y_i suit une loi de Bernoulli et le lien entre l'espérance de y_i et les x_i est effectué grâce à la fonction logit :

$$g\left(\mathbb{E}[y_i]\right) = g(p_\beta(x_i)) = \mathsf{logit}(p_\beta(x_i)) = x_i'\beta.$$

On dit que ces modèles appartiennent à la famille des modèles linéaires généralisés (GLM). Il s'avère que le formalisme GLM (modèle linéaire généralisé) s'applique à d'autres modèles incluant la régression de Poisson. Dans ce chapitre nous présentons tout d'abord la notion de modèle GLM, puis nous étudierons plus en détail la régression de Poisson.

12.1 Le modèle linéaire généralisé (GLM)

Nous commençons par définir un modèle GLM.

Définition 12.1
On considère n observations indépendantes $(x_1, y_1), \ldots, (x_n, y_n)$ où les $x_i \in \mathbb{R}^p$ sont les variables explicatives déterministes et les $y_i \in \mathbb{R}$ sont des variables aléatoires indépendantes à expliquer. Un modèle linéaire généralisé (GLM) est constitué de 3 composantes :

1. **Composante aléatoire.** *Elle consiste à choisir une loi pour la réponse y_i parmi les lois dont la densité se met sous la forme*

$$f_{\alpha_i}(y_i) = \exp\left(\frac{\alpha_i y_i - b(\alpha_i)}{a(\phi)} + c(y_i, \phi)\right), \tag{12.1}$$

où $\alpha_i \in \mathbb{R}$, $\phi \geq 0$ est un paramètre de dispersion et a, b et c sont des fonctions à valeurs dans $\mathbb{R}$.

2. **Composante déterministe** *dénotée $\eta(x_i, \theta)$ est une famille paramétrique de fonctions. Nous considérerons ici la formulation classique et supposerons qu'elle s'exprime sous forme d'une combinaison linéaire des prédicteurs :*

$$\eta(x_i) = \beta_1 x_{i1} + \ldots + \beta_p x_{ip}.$$

3. **Composante de lien.** *Elle spécifie le lien entre les deux composantes, plus précisément le lien entre l'espérance de y_i et la composante déterministe :*

$$g(\mathbb{E}[y_i]) = \eta(x_i) = \beta_1 x_{i1} + \ldots + \beta_p x_{ip}$$

où g est une fonction inversible appelée fonction de lien.

Le problème est de bien choisir les 3 composantes en fonction du problème auquel on est confronté. Il est important de noter que nous transformons la valeur espérée (par la fonction de lien) et non pas les observations.

Choix de la composante aléatoire. Il est généralement guidé par la nature de la variable à expliquer : on pourra utiliser une loi normale (ou Gamma ou Bêta) pour une variable quantitative continue, une loi de Bernoulli pour une variable binaire, une loi multinomiale pour des variables catégoriques avec plusieurs catégories, une loi de Poisson pour une variable de comptage, etc.

Choix de la composante déterministe. C'est le choix le plus difficile pour le statisticien. Il s'agira d'identifier :

— les variables pertinentes à inclure dans le modèle en utilisant par exemple les techniques de choix de variables présentées dans le chapitre 7.
— les éventuelles transformations de variables en utilisant par exemple les résidus partiels présentés dans la section 3.4.
— les interactions pertinentes à ajouter dans le modèle...

Choix de la fonction de lien. En général, la fonction de lien g est bijective et est choisie de sorte que

$$g(\mathbb{E}(Y)) = x'\beta.$$

Il faudra généralement choisir parmi les fonctions de lien classique présentées dans le tableau 12.1.

Nom du lien	Fonction de lien	Domaine
identité	$g(u) = u$	$u \in \mathbb{R}$
log	$g(u) = \log(u)$	$u \in \mathbb{R}^+$
cloglog	$g(u) = \log(-\log(1-u))$	$u \in [0,1]$
logit	$g(u) = \log(u/(1-u))$	$u \in [0,1]$
probit	$g(u) = \Phi^{-1}(u)$	$u \in [0,1]$
réciproque	$g(u) = -1/u$	$u \in \mathbb{R}^+$
puissance	$g(u) = u^\gamma$, $\gamma \neq 0$	$u \in \mathbb{R}^+$

Tableau 12.1 – Fonctions de lien usuelles (Φ désigne la fonction de répartition de la loi $\mathcal{N}(0,1)$).

Pour une composante aléatoire (12.1) fixée, une fonction de lien, appelée *lien canonique*, est souvent privilégiée. C'est le lien défini par $g(u) = (b')^{-1}(u)$ où $b(u)$ est définie à l'équation (12.1).

Remarques

— Les fonctions de lien canonique pour le modèle linéaire gaussien et le modèle logistique sont respectivement les liens identité et logit (voir exercice 12.2).

— La fonction **glm** permet d'ajuster des modèles GLM sur R. Les trois composantes GLM sont à renseigner dans la fonction :

```
glm(formula, data = ..., family = ...(link = ...))
```

On indiquera dans `formula` la composante déterministe : le choix de la combinaison linéaire des variables explicatives. La composante aléatoire et la fonction de lien sont à renseigner dans les arguments `family` et `link`. Par défaut, si l'argument `link` n'est pas renseigné, R utilise alors le lien canonique.

Dans ce chapitre, nous allons nous consacrer à la régression de Poisson qui s'applique quand Y représente des données de comptage. La loi de Poisson est en effet souvent utilisée pour modéliser des variables de comptage, ce sera donc la composante aléatoire du modèle GLM puisqu'il est possible d'écrire la loi d'une variable Y de loi de Poisson de paramètre $\lambda > 0$ comme un cas particulier de 12.1 :

$$\begin{aligned} P(Y = y) &= \frac{\lambda^y}{y!} \exp -\lambda \\ &= \exp(y \log \lambda - \lambda - \log y!), \end{aligned} \tag{12.2}$$

où $y \in \mathbb{N}$, $\alpha = \log(\lambda)$, $b(\alpha) = \exp(\alpha)$, $a(\phi) = 1$ et $c(y, \phi) = -\log y!$. Il est facile de voir que la fonction de lien canonique associée à cette loi est la fonction log (voir exercice 12.2). Par conséquent, pour n observations $(x_i, y_i), i = 1, \ldots, n$ où $x_i \in \mathbb{R}^p$ désigne les variables explicatives et y_i la variable à expliquer à valeurs dans $\mathbb{N}$, la régression de Poisson consistera à modéliser la loi de y_i par une loi de Poisson de paramètre $\lambda_\beta(x_i)$ telle que

$$\log(\mathbb{E}y_i) = \log(\lambda_\beta(x_i)) = x_i'\beta.$$

12.2 Exemple : modélisation du nombre de visites

Afin de prévenir de la malaria, de nombreuses solutions simples et adaptées aux pays en développement existent, par exemple l'utilisation de moustiquaires, ou d'un serpentin dégageant de la fumée... Afin d'évaluer les effets relatifs de ces différentes stratégies, le docteur Perkins a recruté et étudié 1640 enfants de la province de Siaya au Kenya qu'il a suivis pendant de nombreuses années.
Pour chaque enfant, il a enregistré le nombre de visites à l'hôpital local dues à la malaria, l'âge `Age` de la première visite en mois, le sexe `Sexe`, l'`altitude` du domicile de l'enfant, la méthode de prévention `Prev` contre la malaria utilisée à domicile et la durée entre le début du recrutement et la fin de la période d'observation (fin de l'étude ou sortie de l'étude de l'enfant). Le tableau suivant donne les 5 premières observations de l'étude :

Sexe	Age	Altitude	Prev	ObsTime	N.malaria
M	259	1318	Rien	423	1
M	172	1254	Serpentin/Spray	714	6
M	326	1302	Moustiquaire	1080	7
M	573	1301	Moustiquaire	844	0
F	460	1307	Moustiquaire	1082	5

Tableau 12.2 – 5 observations d'enfants.

Après avoir importé les données

```
> Malaria <- read.table("poissonData3.csv", sep = ",", header = T)
```

commençons par résumer les observations :

```
> summary(Malaria)
   Sexe          Age            Altitude            Prev
 F   :815   Min.   :  10.0   Min.   :1129   Autre          :   8
 M   :824   1st Qu.: 219.2   1st Qu.:1266   Moustiquaire   :1100
 NA's:  1   Median : 360.5   Median :1298   Rien           : 457
            Mean   : 419.0   Mean   :1295   Serpentin/Spray:  63
            3rd Qu.: 554.0   3rd Qu.:1320   NA's           :  12
            Max.   :1499.0   Max.   :1515
            NA's   :2        NA's   :105
    Duree          N.malaria
 Min.   :   0.0   Min.   : 0.000
 1st Qu.: 169.8   1st Qu.: 1.000
 Median : 720.0   Median : 3.000
 Mean   : 617.7   Mean   : 4.679
 3rd Qu.:1011.0   3rd Qu.: 7.000
 Max.   :1464.0   Max.   :26.000
```

L'objectif de l'étude consiste à expliquer le nombre de visites (`N.malaria`) par les autres variables (potentiellement) explicatives. Remarquons que la variable `Altitude` contient beaucoup de valeurs manquantes (105) et elle ne sera utilisée que dans un second temps. Nous allons donc utiliser un jeu de données réduit aux trois variables explicatives suivantes `Sexe`, `Prev` et `Duree`. Sur ces données, nous éliminons toutes les valeurs manquantes. Afin d'illustrer la méthode, commençons par expliquer la variable `N.malaria` par la durée d'observation (`Duree`). Débutons cette analyse en représentant les données (fig. 12.1).

```
> plot(N.malaria ~ Duree, data = Malaria, pch = 20, cex = 0.5)
```

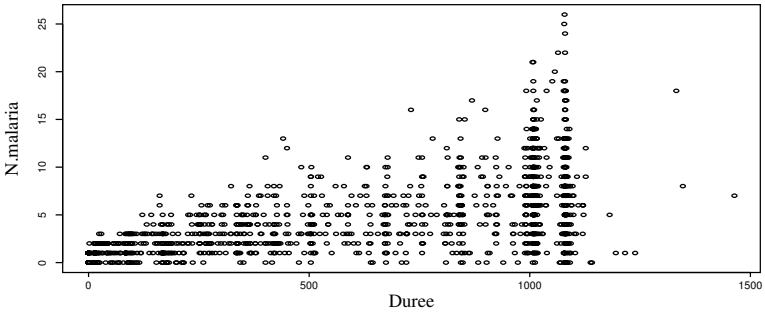


Fig. 12.1 – Variable N.malaria en fonction de `duree`.

Au premier abord, la figure 12.1 laisse penser qu'une régression linéaire simple est possible. Cependant cette approche présente de sérieux inconvénients. La régression des moindres carrés ne garantit pas la positivité de l'espérance de Y et elle ne tient pas compte de de l'hétérogénéité des variances (hétéroscédasticité qui pourrait dépendre de la valeur espérée). Les observations sont clairement plus variables autour de 1000 qu'autour de 100. Enfin, estimer un modèle linéaire en utilisant les moindres carrés n'est pas équivalent à maximiser la vraisemblance et les résultats obtenus ne seront pas optimaux. Malgré cela, nous pouvons tout de même calculer et ajouter la droite de régression simple (fig. 12.2)

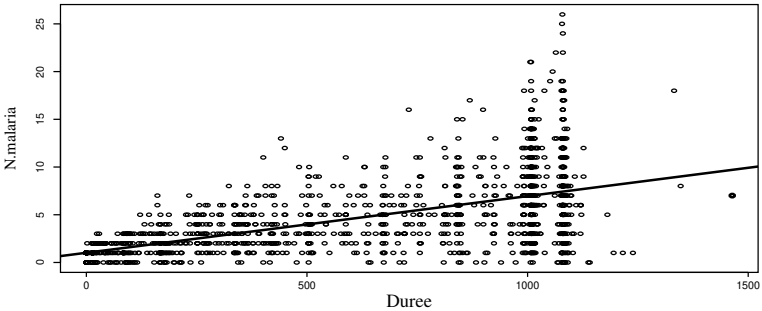


Fig. 12.2 – Variable N.malaria en fonction de l'âge et droite des MC.

en utilisant les codes suivants :

```
> mod.lin <- lm(N.malaria ~ Duree, data = Malaria)
> abline(a=coef(mod.lin)[1],b=coef(mod.lin)[2],lwd=2)
```

La variable à expliquer (`N.malaria`) est une variable de comptage : il s'agit du nombre de visites médicales pour la malaria. Nous allons utiliser le même raisonnement que dans le chapitre 11 (p. 251) où Y était binaire. Nous avions utilisé la loi de Bernoulli de paramètre $p_\beta(x)$ pour Y et avons lié cette loi à une variable explicative X selon $\mathsf{logit}(p_\beta(x)) = \beta_1 + \beta_2 x$. La variable Y étant une variable de comptage, il est naturel d'utiliser la loi de Poisson. Cette dernière dépend d'un paramètre λ qui est l'espérance de la loi. Chercher à expliquer Y par la durée revient à se demander si le paramètre $\lambda > 0$ varie lorsque la durée varie et donc à considérer le paramètre $\lambda(x)$ comme une fonction de la durée x en considérant

$$Y \sim \mathcal{P}(\lambda_\beta(x)), \tag{12.3}$$

où $\lambda_\beta = \mathbb{E}(Y)$ est une fonction à spécifier.
Nous avons vu dans la section 12.1 que la fonction de lien canonique pour la loi de Poisson est la fonction log. Ainsi, pour cet exemple, on lie l'espérance de Y (`N.malaria`) à la variable explicative X (`duree`) via

$$\log(\lambda_\beta(x)) = \beta_1 + \beta_2 x.$$

On a ainsi $\lambda_\beta(x) = \exp(\beta_1 + \beta_2 x)$, ce qui nous permet de garantir la positivité de $\lambda_\beta(x)$. Ce modèle est un modèle linéaire généralisé (section 12.1). On utilisera donc la fonction **glm**, avec l'option `family=poisson` pour estimer ses paramètres (souhaitant utiliser le lien canonique, on ne le renseigne pas puisqu'il sera choisi par défaut).

```
> modP <- glm(N.malaria ~ Duree, data = Malaria, family = poisson)
> modP

Call:  glm(formula = N.malaria ~ Duree, family = poisson,
                                data = Malaria)

Coefficients:
(Intercept)        Duree
   0.429459     0.001508

Degrees of Freedom: 1626 Total (i.e. Null);  1625 Residual
Null Deviance:            5710
Residual Deviance: 3325         AIC: 8125
```

Nous obtenons les estimateurs des paramètres du modèle (12.3) et nous pouvons ajouter la courbe $\lambda_{\hat\beta}(x) = \exp(0.429459 + 0.001508x)$ en pointillé sur le graphe précédent et obtenir la figure 12.3.

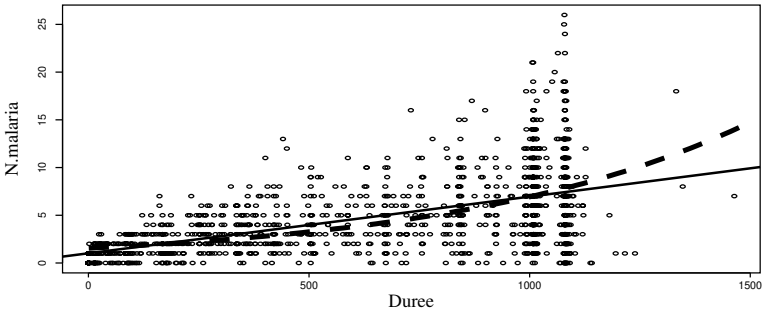


Fig. 12.3 – Variable N.malaria en fonction de `duree`, droite des MC (trait plein) et ajustement de Poisson (tirets).

Sur ce premier exemple, nous voyons la différence entre une régression MC classique et une régression de Poisson. La régression de Poisson permet de tenir compte de l'aspect positif de la variable de comptage Y. De plus, la loi de Poisson $\mathcal{P}(\lambda_\beta(x))$ a pour variance $\lambda_\beta(x)$ qui dépend de x. Cela permet de tenir compte de l'augmentation de variance avec la durée. Nous allons maintenant étudier plus en détail ce modèle et le généraliser à plusieurs variables explicatives (quantitatives et ou qualitatives).

12.3 Régression Log-linéaire

Il est usuel lorsque nous traitons de données de comptage de faire les hypothèses simplificatrices suivantes :

1. les nombres d'infections dues à la malaria dans des intervalles de temps disjoints sont indépendants ;

2. le taux d'infection est constant dans le temps et est noté λ ;

3. la probabilité d'avoir 2 (ou plus) infections dans un petit intervalle de temps Δ tend vers 0 quand la longueur de l'intervalle tend vers 0.

Sous ces conditions, le nombre d'infections dues à la malaria N suit une loi de Poisson de paramètre λ.

12.3.1 Le modèle

Le modèle de Poisson, ou régression log-linéaire ou encore régression de Poisson, permet d'expliquer une variable de comptage Y par p variables explicatives $X_1, \ldots, X_p$. Nous le définissons ci-dessous.

Définition 12.2
Soit $(x_1, y_1), \ldots, (x_n, y_n)$ *n observations telles que* $x_i = (x_{i1}, \ldots, x_{ip}) \in \mathbb{R}^p$ *et* $y_i \in \mathbb{N}$. *Le modèle de régression de Poisson suppose que les observations* $y_i, i = 1, \ldots, n$

sont des réalisations de variables aléatoires indépendantes et de loi de Poisson de paramètre $\lambda_\beta(x_i)$ vérifiant

$$\log(\lambda_\beta(x_i)) = \beta_1 x_{i1} + \ldots + \beta_p x_{ip} = x_i'\beta. \tag{12.4}$$

Remarques

— Comme pour les modèles linéaire et logistique, si on souhaite inclure la variable constante dans le modèle, on prendra x_{i1} égal à 1 pour tout i variant de 1 à n.

— Les variables explicatives peuvent être quantitatives et/ou qualitatives. Pour des variables qualitatives, R effectue un codage disjonctif complet et utilise comme contraintes identifiantes que le coefficient associé à la première modalité (par ordre alphabétique) de chaque variable qualitative vaut zéro. Si nous considérons par exemple un modèle avec deux variables explicatives : X_1 quantitative et X_2 qualitative qui prend ses valeurs dans $\{A, B, C\}$, alors le modèle de Poisson s'écrit

$$\log(\lambda_\beta(x)) = \beta_0 + \beta_1 x_1 + \beta_2 \mathbf{1}_A(x) + \beta_3 \mathbf{1}_B(x) + \beta_4 \mathbf{1}_C(x)$$

muni de la contrainte $\beta_2 = 0$. Les problèmes d'identifiabilité ont été abordés pour les modèles linéaire et logistique. Ils se traitent exactement de la même façon pour la régression de Poisson. Nous renvoyons donc le lecteur au chapitre 6 et à la section 11.1.3 pour plus de détails.

12.3.2 Estimation

On dispose de n observations $(x_1, y_1), \ldots, (x_n, y_n)$ où $(x_i, y_i) \in (\mathbb{R}^p \times \mathbb{N})$ et les y_i sont des réalisations de variables aléatoires indépendantes de loi de Poisson de paramètre $\lambda_\beta(x_i), i = 1, \ldots, n$. La vraisemblance du modèle de Poisson s'écrit donc

$$L(Y, \beta) \quad = \quad \prod_{i=1}^{n} \frac{\lambda_\beta(x_i)^{y_i}}{y_i!} \exp\left(-\lambda_\beta(x_i)\right),$$

avec $Y = (y_1, \ldots, y_n)$ et $\beta = (\beta_1, \ldots, \beta_p)$. Il est d'usage de considérer la log-vraisemblance

$$\begin{aligned}\mathcal{L}(Y, \beta) &= \sum_{i=1}^{n} y_i \log(\lambda_\beta(x_i)) - \lambda_\beta(x_i) - \log(y_i!) \\ &= \sum_{i=1}^{n} y_i (x_i^t \beta) - \exp(x_i^t \beta) - \log(y_i!). \end{aligned} \tag{12.5}$$

Le dernier terme $(-\log(y_i!))$, ne dépend pas de β, n'influera pas dans la recherche du maximum de la (log-)vraisemblance et pourra ne pas apparaître dans certaines équations. Les estimateurs du maximum de vraisemblance, s'ils existent [1], sont solution des équations normales

$$S(\beta) = \nabla \mathcal{L}(\beta) = \sum_{i=1}^{n} (y_i - \exp(x_i^t \beta)) x_i = X'(Y - \Lambda_\beta) = 0, \tag{12.6}$$

1. Les EMV pour β n'existent pas si pour des $j \in \{1, \cdots, p\}$, $|\hat{\beta}_j| = \infty$

où $\Lambda_\beta = (\lambda_\beta(x_1), \ldots, \lambda_\beta(x_n))$. La fonction $S(.)$ est appelée *fonction de score*. Nous retrouvons le même type de formulation qu'avec le modèle logistique (voir eq. 11.9 p. 258). En général, il n'existe pas de solution analytique à l'équation (12.6). En se basant sur la matrice hessienne

$$\nabla^2 \mathcal{L}(\beta) = -X' W_\beta X \quad \text{avec} \quad W_\beta = \text{diag}(\lambda_\beta(x_1), \ldots, \lambda_\beta(x_n)),$$

on montre que l'opposé de la log-vraisemblance $\beta \mapsto -\mathcal{L}(Y, \beta)$ est strictement convexe (si un des $x_i \neq 0$, ce qui est en général le cas...). Cette fonction est de plus en général une fonction coercive [2], ce qui nous garantit l'existence d'un $\hat{\beta}$ unique minimiseur de $-\mathcal{L}(Y, \beta)$ sur $\mathbb{R}^p$.
Comme dans le chapitre précédent, les techniques numériques classiques d'optimisation peuvent être appliquées ici afin de trouver l'unique maximum. Pour cela, il suffit d'utiliser les résultats de la section 11.2.2 (p. 259) avec l'équation du score (12.6). L'algorithme IRLS (algorithme 4 p. 260) est utilisé avec la matrice

$$A(\beta) = \nabla^2 \mathcal{L}(Y, \beta) = -X' W_\beta X$$

par la fonction **glm** de R pour calculer l'estimateur du maximum de vraisemblance.

12.3.3 Tests et intervalles de confiance

Les principales propriétés de l'EMV découlent de la théorie du maximum de vraisemblance. Sous des hypothèses similaires à celles du théorème 11.1, l'EMV existe et est consistant. De plus, pour n assez grand, il est approximativement gaussien, sans biais et sa matrice de variance covariance est proche de l'inverse de la matrice d'information de Fisher donnée par

$$\mathcal{I}_n(\beta) = -\mathbb{E}\left[\nabla^2 \mathcal{L}(Y, \beta)\right] = X' W_\beta X.$$

Plus précisément, la statistique

$$(\hat{\beta} - \beta)' \mathcal{I}_n(\hat{\beta})(\hat{\beta} - \beta)$$

converge en loi vers une loi du χ^2 à p degrés de liberté. Si on désigne par $\hat{\sigma}_j^2$ le j^{e} terme de la diagonale de $(X' W_{\hat{\beta}} X)^{-1}$, on déduit que

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma}_j} \tag{12.7}$$

converge vers la loi normale centrée réduite. On pourra déduire de ce résultat des intervalles de confiance ainsi que des procédures de test sur les paramètres. On rejettera par exemple l'hypothèse nulle du test

$$\text{H}_0 : \beta_j = 0 \quad \text{contre} \quad \text{H}_1 : \beta_j \neq 0$$

2. Par définition $\beta \mapsto -\mathcal{L}(Y, \beta)$ si elle vérifie $\lim_{\|\beta\| \to +\infty} -\mathcal{L}(Y, \beta) = +\infty$

au niveau $\alpha \in]0,1[$ si $|\hat{\beta}_j/\hat{\sigma}_j|$ dépasse le quantile d'ordre $1-\alpha/2$ de la loi normale centrée réduite.
Reprenons l'exemple de la malaria. Nous cherchons maintenant à expliquer le nombre de visites à l'hôpital par les variables `Duree`, `Sexe` et `Prev`. Les paramètres du modèle sont toujours estimés avec la fonction **glm**.

```
> modP3 <- glm(N.malaria ~ Duree + Sexe + Prev, data = Malaria,
+         family = poisson )
> summary(modP3)
Call:
glm(formula = N.malaria ~ Duree + Sexe + Prev, family=poisson,
    data = Malaria)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.0903  -0.9488  -0.3503   0.7025   5.0166

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         0.1623101  0.1803505   0.900   0.3681
Duree               0.0015101  0.0000343  44.031   <2e-16 ***
SexeM               0.0550890  0.0229690   2.398   0.0165 *
PrevMoustiquaire    0.2432678  0.1774693   1.371   0.1704
PrevRien            0.2255828  0.1781379   1.266   0.2054
PrevSerpentin/Spray 0.2452247  0.1851801   1.324   0.1854
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 5710.4  on 1626  degrees of freedom
Residual deviance: 3317.3  on 1621  degrees of freedom
AIC: 8124.6

Number of Fisher Scoring iterations: 5
```

La dernière ligne indique le nombre d'itérations nécessaire à la convergence de l'algorithme de maximisation de la log-vraisemblance. Le tableau `Coefficients` renvoie des informations sur les paramètres du modèle. Ce dernier est composé d'une variable quantitative et de deux variables qualitatives qui prennent 2 et 4 modalités et s'écrit

$$\begin{aligned}\log(\lambda_\beta(x)) &= \beta_0 + \beta_1 \text{Duree} + \beta_2 \mathbf{1}_{\text{M}}(\text{Sexe}) \\ &\quad + \beta_3 \mathbf{1}_{\text{Moust}}(\text{Prev}) + \beta_4 \mathbf{1}_{\text{Rien}}(\text{Prev}) + \beta_5 \mathbf{1}_{\text{Serp}}(\text{Prev}).\end{aligned} \tag{12.8}$$

Comme pour les modèles linéaire et logistique, on remarquera que des contraintes

identifiantes sont utilisées pour les variables qualitatives (voir par exemple section 11.1.3). Ici les coefficients pour les modalités `F` et `Autre` des variables `Sexe` et `Prev` sont imposés égaux à 0. Ce choix est fait par défaut par R car ce sont les premières modalités dans l'ordre alphabétique [3].
La première colonne de la partie `Coefficients` du résumé (colonne intitulée `Estimate`) nous donne les estimations des paramètres. Rappelons que le coefficient constant (`Intercept`) est proposé par défaut. Le coefficient associé à `Duree` est estimé à 0.0015. Le coefficient pour le sexe `M` est estimé à 0.055089. Le fait de passer du sexe `F` au sexe `M` donne un accroissement à $\log(\lambda(x))$ de 0.055089, c'est-à-dire qu'en moyenne le fait d'être de sexe masculin augmente le nombre de visites de $\exp(0.055089) \approx 1.06$. En termes d'interprétation nous pourrions donc proposer que les individus masculins sont plus faibles en termes de résistance à la malaria ou que les individus masculins sont, dans cette région du monde, l'objet de plus de soins que les féminins.
La seconde colonne (`Std. Error`) renvoie les écarts-types $\hat{\sigma}_j$ des estimateurs. La valeur 0.00003 pour le paramètre correspondant à la variable `Duree` signifie que la valeur estimée de 0.0015 est hautement significative comme on peut le voir grâce aux deux dernières colonnes du tableau : la colonne `z value` présente les valeurs des statistiques (l'estimateur divisé par son écart-type) de test tandis que la dernière colonne `Pr(>|z|)` renseigne les probabilités critiques.
Sur cet exemple, pour un niveau $\alpha = 5\%$ on acceptera la nullité des paramètres associés à la variable `Prev` (paramètre β_3, β_4 et β_5 dans (12.8)).
Concernant les moyens de prévention, le premier moyen de prévention (`Autre`) est imposé à 0. Les coefficients estimés sont donc interprétables comme l'écart à cette modalité. En termes d'interprétation ce n'est absolument pas confortable et il serait plus facile de considérer l'écart d'un moyen de prévention à aucun moyen de prévention. Il faut donc changer la contrainte identifiante en imposant que le coefficient associé à la modalité `Rien` soit égal à 0 :

```
> Malaria$Prev <- relevel(Malaria$Prev, ref = "Rien")
> modP3 <- glm(N.malaria ~ Duree + Sexe + Prev, data = Malaria,
+         family = poisson )
> summary(modP3)
Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)           0.3878929  0.0389800   9.951   <2e-16 ***
Duree                 0.0015101  0.0000343  44.031   <2e-16 ***
SexeM                 0.0550890  0.0229690   2.398   0.0165 *
PrevAutre            -0.2255828  0.1781379  -1.266   0.2054
PrevMoustiquaire      0.0176850  0.0255967   0.691   0.4896
PrevSerpentin/Spray   0.0196420  0.0590690   0.333   0.7395
```

Nous ne mettons ici que la partie `Coefficients`, le reste n'a pas changé. Bien

3. Plus précisément, il s'agit du premier niveau des facteurs tel qu'il apparaît dans l'appel à la fonction **levels** ; voir aussi **relevel** pour changer.

évidemment les coefficients associés à `SexeM` et `Duree` n'ont pas changé.
Ainsi, si nous voulons analyser l'effet d'utiliser comme moyen de prévention un spray, alors nous pouvons dire que ce moyen de prévention entraîne en moyenne $\exp(0.0245) = 1.02$ fois plus de visites que ne rien faire. Cependant, l'écart-type est de 0.18 et le test accepte la nullité du paramètre associé au moyen de prévention `Spray`. De plus, l'utilisation de spray montre que la famille est sensibilisée au problème de la malaria (ou financièrement plus aisée) et cela peut contribuer aussi à augmenter le nombre de visites.
Nous voyons sur les deux dernières sorties présentées que les tests de nullité des paramètres ne sont pas les mêmes (les probabilités critiques ne sont pas identiques). En effet, en présence de variables qualitatives, ces tests dépendent de la contrainte identifiante et ne sont pas toujours pertinents. Il est souvent préférable de s'intéresser à l'influence globale de la variable qualitative sur la variable à expliquer. Sur notre exemple, on peut par exemple se demander si le sexe ou le moyen de prévention a une influence sur le nombre de visites. Dire que la prévention n'a pas d'influence revient à dire que les coefficients du modèle associé à la variable `Prev` ont tous la même valeur. On peut traduire cela dans le modèle (12.8) par les coefficients $\beta_3, \beta_4, \beta_5$ sont nuls puisque la contrainte utilisée fixe le coefficient de la modalité `Autre` à 0. Ainsi dans ce cas, nous sommes amenés à tester

$$\mathrm{H}_0 : \beta_3 = \beta_4 = \beta_5 = 0 \quad \text{contre} \quad \mathrm{H}_1 : \exists j \in \{3, 4, 5\} : \beta_j \neq 0.$$

Il ne s'agit plus de tester la nullité d'un paramètre mais celle d'un sous-groupe de paramètres. Pour cela, plusieurs statistiques de test peuvent être utilisées. On peut par exemple utiliser celle du rapport de vraisemblance

$$-2(\mathcal{L}_{\mathrm{H}_0}(Y, \hat{\beta}_{\mathrm{H}_0}) - \mathcal{L}(Y, \hat{\beta})),$$

où $\mathcal{L}_{\mathrm{H}_0}(Y, \hat{\beta}_{\mathrm{H}_0})$ désigne la log-vraisemblance du modèle sous H_0, c'est-à-dire la log-vraisemblance du modèle sans la variable `Prev`. Cette statistique converge en loi vers une loi de χ^2 à $p_1 - p_0$ degrés de liberté avec p_0 et p_1 les nombres de paramètres du modèle complet et du modèle sous H_0. Dans notre exemple $p_0 = 3$ et $p_1 = 5$. Sous R on peut obtenir la statistique de test avec

```
> modP2 <- glm(N.malaria ~ Duree + Sexe, data = Malaria,
                            family = poisson)
> -2*(logLik(modP2)-logLik(modP3))
'log Lik.' 2.448823 (df=3)
```

Pour un test au niveau $\alpha = 5\%$, cette valeur est à comparer avec le quantile d'ordre 0.95 de la loi du χ^2 à 3 degrés de liberté

```
> qchisq(0.95,df=3)
[1] 7.814728
```

La statistique de test étant inférieure au quantile, on conservera l'hypothèse nulle et on pourra donc conclure que le mode de prévention n'a pas d'effet sur le nombre

de visites. On peut bien entendu effectuer ce test directement à l'aide de la fonction **anova** :

```
> anova(modP2, modP3, test = "LRT")
Analysis of Deviance Table

Model 1: N.malaria ~ Duree + Sexe
Model 2: N.malaria ~ Duree + Sexe + Prev
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      1624     3319.8
2      1621     3317.3  3   2.4488   0.4846
```

Remarque
La fonction **Anova** du package **car** permet de tester l'effet de toutes les variables dans le modèle dans l'ordre spécifié à l'intérieur de la formule. Changer l'ordre changera les valeurs présentées et peut même impacter la signification des tests. Par exemple :

```
> library(car)
> Anova(modP3, test = "LR")
Analysis of Deviance Table (Type II tests)

Response: N.malaria
           LR Chisq Df Pr(>Chisq)
Duree       2380.25  1    < 2e-16 ***
Sexe           5.75  1    0.01645 *
Prev     2.45  3    0.48461
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Nous avons ici les résultats des tests de rapport de vraisemblance pour les trois variables du modèle. On retrouve bien entendu le même résultat que précédemment pour `Prev`. Concernant le sexe, on pourra considérer, au niveau 5% qu'il a également un effet sur le nombre de visites ainsi que pour la variable `Duree`.
Nous nous sommes restreints au test du rapport de vraisemblance, d'autres statistiques peuvent être utilisées pour tester la nullité d'un sous-ensemble de paramètres, comme la statistique de Wald. Cette statistique est définie de la même façon que pour le modèle logistique. Nous renvoyons le lecteur à la section 11.3.2 pour la réalisation de ce test.

Concernant les intervalles de confiance, on déduit de l'approximation de la loi de (12.7) par la loi $\mathcal{N}(0,1)$ que

$$IC_{1-\alpha}(\beta_j) = \left[\hat{\beta}_j - u_{1-\alpha/2}\hat{\sigma}_j; \hat{\beta}_j + u_{1-\alpha/2}\hat{\sigma}_j\right]$$

est un intervalle de confiance (asymptotique) de niveau $1-\alpha$ pour β_j. On rappelle

que $u_{1-\alpha/2}$ désigne le quantile d'ordre $1-\alpha/2$ de la loi $\mathcal{N}(0,1)$. On obtiendra ces intervalles avec la fonction **confint.default** :

```
>  round(confint.default(modP3),3)
                    2.5 % 97.5 %
(Intercept)          0.311  0.464
Duree                0.001  0.002
SexeM                0.010  0.100
PrevAutre           -0.575  0.124
PrevMoustiquaire    -0.032  0.068
PrevSerpentin/Spray -0.096  0.135
```

Comme pour le modèle logistique, la fonction **confint** permet d'obtenir également des intervalles de confiance :

```
> round(confint(modP3),3)
                    2.5 % 97.5 %
(Intercept)          0.311  0.464
Duree                0.001  0.002
SexeM                0.010  0.100
PrevAutre           -0.596  0.105
PrevMoustiquaire    -0.032  0.068
PrevSerpentin/Spray -0.098  0.134
```

Ces derniers sont calculés à partir de la vraisemblance (voir exercice 11.9 pour plus de détails).

12.3.4 Choix de variables

Tout comme pour les modèles linéaire et logistique, sélectionner les variables est souvent une étape importante pour construire un modèle de Poisson. En effet, lorsque le nombre de variables est grand, il est important de supprimer les variables non influentes pour augmenter la précision des estimateurs et pour mieux interpréter le modèle. Les techniques permettant de sélectionner les variables dans un modèle de Poisson sont standards. Elles consistent le plus souvent à construire des modèles basés sur des sous-ensembles de l'ensemble des variables et à les comparer à l'aide de critères de choix de modèles de type AIC ou BIC défini dans l'équation (11.24). Lorsque le nombre de variables p est relativement petit (inférieur à 50), on peut utiliser des approches exhaustives qui consistent à construire tous les modèles possibles à partir de ces p variables (il y en a 2^{p-1}) et à choisir celui qui optimise le critère choisi.

Considérons le jeu de données `Malaria` complet, dans lequel on enlève les individus qui contiennent des données manquantes.

```
> Malaria <- read.table("poissonData.csv", sep=",", header=T)
> Malaria1 <- na.omit(Malaria)
```

Sur R, on peut utiliser le package **bestglm** pour effectuer une procédure de choix de variables exhaustives.

```
 > mod_sel <- bestglm(Malaria1, family = poisson)
 > mod_sel$BestModels
   Sexe   Age Altitude    Prev Duree Criterion
1 FALSE  TRUE     TRUE   FALSE  TRUE  7384.946
2 FALSE FALSE     TRUE   FALSE  TRUE  7387.814
3  TRUE  TRUE     TRUE   FALSE  TRUE  7390.053
4  TRUE FALSE     TRUE   FALSE  TRUE  7393.119
5 FALSE  TRUE    FALSE   FALSE  TRUE  7401.021
```

Le critère utilisé par défaut est le BIC (dernière colonne). On obtient ici les 5 meilleurs modèles au sens de ce critère. Le modèle sélectionné aura une valeur de BIC de 7385 et contiendra les variables `Age` `Altitude` et `Duree`. Pour ce critère, le mode de prévention n'apparaît pas comme une variable pertinente.
Cette procédure exhaustive nécessite l'ajustement de 2^{p-1} modèles. Elle peut se révéler coûteuse en temps de calcul lorsque p est grand. Dans ce cas, on a souvent recours aux procédures pas à pas présentées dans la section 7.4.2. Sur R, on pourra utiliser la fonction **step** ou modifier l'argument `method` dans la fonction **bestglm** pour utiliser ces procédures.

12.4 Exercices

Exercice 12.1 (Questions de cours)

1) Nous souhaitons effectuer une régression dont la variable à expliquer est une variable de comptage, on utilise la fonction **glm** en précisant
   A. family=binomial,
   B. en ne précisant rien,
   C. family=poisson.
2) Lors d'une régression de poisson (hors modèle saturé ou constant), les estimateurs sont obtenus en utilisant un algorithme itératif :
   A. oui toujours,
   B. non jamais,
   C. seulement si les variables explicatives sont qualitatives.
3) Un estimateur de la variance de $\hat{\beta}$ de β dans le cas de la régression de Poisson vaut :
   A. $\hat{\sigma}^2(X'X)^{-1}$;
   B. $(X'W_{\hat{\beta}}X)^{-1}$;
   C. $(W_\beta)^{-1}$.
4) Pour estimer les paramètres d'une régression de poisson, on
   A. maximise la vraisemblance,
   B. maximise la déviance,
   C. minimise les moindres carrés.

5) Les espérances des effectifs (notés λ_t) du modèle « saturé » en chaque unique point du design $\tilde{x}_t$ (avec $\tilde{x}_t \neq \tilde{x}_l$ pour $t \neq l$) sont estimées par
   A. par la moyenne $\frac{1}{n_t}\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$,
   B. par la somme $\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$,
   C. par le logarithme népérien de la moyenne $\log \frac{1}{n_t}\sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$,
   D. par le logarithme népérien de la somme $\log \sum_{i=1}^{n_t} y_i \mathbb{1}_{x_t}(x_i)$ avec $n_t = \sum_{i=1}^{n_t} \mathbb{1}_{x_t}(x_i)$.

6) Les modèles de régression de Poisson imposent
   A. que les y_i sont iid et suivent une loi de Bernoulli,
   B. que les y_i sont iid et suivent une loi de Poisson,
   C. que les y_i sont iid et suivent une loi normale.

7) Les modèles de régression de Poisson imposent
   A. qu'en chaque unique point du design la variance est constante (égale à σ^2),
   B. qu'en chaque unique point du design la variance vaut $n_t p_t(1-p_t)$,
   A. qu'en chaque unique point du design la variance vaut λ_t.

8) Les intervalles de confiances en régression de Poisson (de niveau $1-\alpha$) pour les coordonnées de β
   A. sont approximativement de niveau $1-\alpha$,
   B. sont précisément de niveau $1-\alpha$.

Exercice 12.2
Calculer les fonctions de lien canonique pour le modèle linéaire gaussien, le modèle logistique et le modèle de Poisson.

Exercice 12.3
Soit X une variable aléatoire suivant une loi de Poisson $\mathcal{P}(\lambda)$.

1) Montrez que les moments factoriels d'ordre r d'une variable aléatoire suivant une loi de Poisson valent

$$\mathbb{E}[X(X-1)\dots(X-r+1)] \quad = \quad \lambda^r.$$

2) Déduire que l'espérance et la variance valent λ.

Exercice 12.4 (Stabilisation de la variance)
Vérifier graphiquement que si X suit une loi de Poisson, alors $Z = \sqrt{X}$ a comme variance environ 1/4 (on pourra prendre un échantillon de taille 1000000 et calculer sa variance empirique pour des valeurs de λ variant entre 1 et 20.

Exercice 12.5 (Stabilisation de la variance)
Montrer que $\mathrm{V}(\sqrt{X}) \approx \frac{1}{4}$.

Exercice 12.6
Reprenez les données de la malaria.

1) Représenter graphiquement les barres suivantes : en abscisse les nombres de visites et en ordonnée le nombre total de visites pour les femmes (ordonnée positive) ou pour les hommes (ordonnée négative). Vous devriez obtenir le graphique suivant

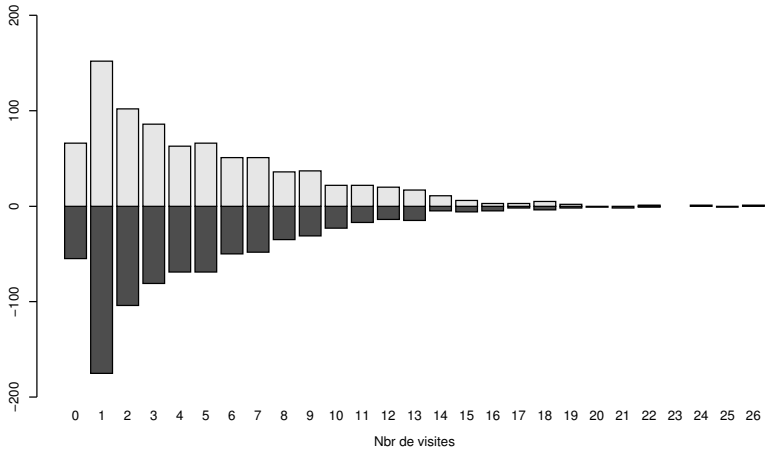


Fig. 12.4 – Représentation des données et effet du sexe

avec le code

```
> tab <- lapply( split( Malaria$N.malaria, Malaria$Sexe ), table )
> Tab <- matrix( 0,2, max( Malaria$N.malaria )+1 )
> colnames(Tab) <- 0:max( Malaria$N.malaria )
> Tab[1, names(tab[[1]]) ] <- tab[[1]]
> Tab[2, names(tab[[2]]) ] <- tab[[2]]
> barplot( Tab, offset = -Tab[1,])
```

Commenter le code ci-dessus.
2) Comparer la moyenne du nombre de visites pour cause de malaria à l'hôpital pour les hommes et pour les femmes. Pour des données de comptage, on considère souvent le rapport des moyennes ou de façon équivalente la différence des logarithmes des moyennes.
3) Y a-t-il une différence entre le logarithme du nombre de visites à l'hôpital pour cause de malaria entre les filles et les garçons ? Un code possible est

```
> mm <- sapply( split( Malaria$N.malaria, Malaria$Gender ),mean,na.rm=T)
> round( c( log( mm[1] ), diff( log( mm ) ) ), 5)
```

4) Retrouver ce résultat en effectuant une régression de Poisson avec comme seule variable explicative la variable `Sexe`.

Exercice 12.7 (Table de contingence et loi de Poisson)
L'analyse de table de contingence qui compte le nombre d'occurrences entre 2 (ou plus) variables qualitatives est un cas spécifique de variables de comptage. Nous allons considérer l'analyse entre les variables `Prev` et `Sexe` des données de Malaria. L'objectif de cet exercice est de montrer qu'il est possible d'utiliser la régression de poisson pour analyser un tableau de contingence.
1) Avec la fonction **table()**, donner le tableau de contingence.
2) Effectuer un test du χ^2 en utilisant **chisq.test()**

3) Créer un data-frame en mettant en colonne les effectifs de chaque cellule et les variables `Prev` et `Sexe`. Le résultat souhaité figure ci-dessous :

```
    Y Sexe      Prev
1   2    F           Autre
2   6    M           Autre
3 557    F    Moustiquaire
4 543    M    Moustiquaire
5 223    F            Rien
6 233    M            Rien
7  28    F Serpentin/Spray
8  35    M Serpentin/Spray
```

4) Ajuster une régression de Poisson avec interaction et sans coefficient constant

```
> mod1 <- glm(Y ~ -1 + Sexe:Prev, data = don, family = poisson)
```

— Y-a-t-il des contraintes identifiantes ?
— Le modèle est-il saturé ?
— Retrouvez les estimations des paramètres de `mod1`.

5) Ajuster une régression de Poisson sans interaction

```
> mod2 <- glm(Y ~ 1 + Sexe + Prev, data = don, family = poisson)
```

— Y-a-t-il des contraintes identifiantes ?
— Le modèle est-il saturé ?
— Interprétez les estimations de `mod2`.

6) Comparer les 2 modèles en utilisant l'AIC, quel modèle conservez-vous ? Ce résultat est-il surprenant connaissant le résultat de la question 2 ?

Exercice 12.8 (Table de contingence et probabilité)
Considérons une table de contingence à 2 entrées et notons π_{ij} la probabilité d'être dans la cellule (i, j). Ainsi dans l'exemple de l'exercice 12.7, nous avons

Sexe/Prévention	Autre	Moustiquaire	Rien	Serpentin/Spray	ligne
F	π_{11}	π_{12}	π_{13}	π_{14}	$\pi_{1.}$
M	π_{21}	π_{22}	π_{23}	π_{24}	$\pi_{2.}$
colonne	$\pi_{.1}$	$\pi_{.2}$	$\pi_{.3}$	$\pi_{.4}$	1

Tableau 12.3 – Probabilités

1) En utilisant la formule des probabilités totales, retrouvez que la probabilité d'être une femme notée $\pi_{1.}$ vaut $\sum_{j=1}^{4} \pi_{ij}$.
2) Retrouvez de même les probabilités des marges (lignes et colonnes).
3) Pouvez-vous écrire les contraintes associées aux lignes ? aux colonnes ?
4) Si les 2 variables sont indépendantes, écrivez π_{ij} en fonction de $\pi_{i.}$ et de $\pi_{.j}$,. On décrit l'écart au modèle en notant

$$\pi_{ij} = (\pi_{i.}\pi_{.j}) \left(\frac{\pi_{ij}}{\pi_{i.}\pi_{.j}} \right).$$

5) Le tableau de contingence reporte des effectifs par cellule, l'effectif total sera noté N. En multipliant l'équation précédente par N, en notant $\lambda_{ij} = N\pi_{ij}$ et en passant au $\log$, montrez que

$$\log \lambda_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij}.$$

Indiquez précisément les différents paramètres (par exemple $\alpha_i = \log \pi_{i.}$).
6) Retranscrivez les contraintes initiales associées aux lignes et colonnes en contrainte identifiantes. En remarquant que λ_{ij} est un effectif moyen, on retrouve la modélisation de Poisson vue à l'exercice 12.7 avec des contraintes spécifiques.

Exercice 12.9 (Loi Multinomiale)
La loi multinomiale permet de modéliser une variable qualitative à K modalités en modélisant les probabilités d'obtenir des effectifs $n_1, \ldots, n_k$ pour chaque modalité selon :

$$\Pr(N_1 = n_1, \ldots, N_K = n_K) = \frac{n!}{n_1! \ldots n_K!} \prod_{k=1}^{K} \pi_k^{n_k}.$$

Nous avons que π_k représente la probabilité d'obtenir la modalité k et donc $\sum_k \pi_k = 1$. De plus, $\sum_k n_k = n$. La loi est classiquement notée $\mathcal{M}(n, \pi_1, \ldots, \pi_K)$.
Montrer qu'une variable distribuée selon une loi Multinomiale appartient à la famille exponentielle et que la fonction de lien naturelle est la fonction $\log$.

Exercice 12.10
Reprenons un exemple classique cité par Noël (2015) étudiant les troubles de conduites alimentaires chez les adolescentes. Les données sont présentées dans le tableau 12.4. La question est la suivante :« peut-on dire que la distribution des intentions est la même pour les deux groupes ? ».

`Groupe/Intention`	Maigrir	Garder	Grossir
Afro-Américain	$n_{11} = 47$	$n_{12} = 28$	$n_{13} = 24$
Européen	$n_{21} = 352$	$n_{22} = 152$	$n_{23} = 31$

Tableau 12.4 – Désir de changer de poids chez les adolescentes

La loi Multinomiale est définie à l'exercice 12.9.
1) Si la variable `Intention` est la même dans chaque groupe alors les observations $i \in \{1, 2\}$ sont indépendantes

$$\Pr(N_{11} = n_{11}, \ldots, N_{23} = n_{23}) = \prod_{i=1}^{2} \Pr(N_{i1} = n_{i1}, N_{i2} = n_{i2}, N_{i3} = n_{i3})$$

et montrez que pour l'observation i on a

$$\Pr(N_{i1} = n_{i1}, N_{i2} = n_{i2}, N_{i3} = n_{i3}) = \frac{n_{i.}!}{n_{i1}! n_{i1}! n_{i3}!} \prod_j (\pi_{.j})^{n_{ij}}.$$

2) Proposez une autre modélisation multinomiale dans le cas où `Intention` diffère dans chaque `Groupe`.

3) Ecrire les log-vraisemblances des deux modèles sans oublier les contraintes sur les paramètres. Réécrire les log-vraisemblances avec le changement de variable suivant pour les probabilités :

$$\pi_{ij} = \frac{\exp \eta_{ij}}{\sum_j \exp \eta_{ij}}, \quad \text{avec} \begin{cases} \eta_{ij} & = \gamma_{ij} \text{ si on retient un effet } \texttt{Groupe} \\ \eta_{ij} & = \mu_j \text{ si on ne retient pas d'effet } \texttt{Groupe} \end{cases}$$

4) Ecrire une modélisation de Poisson sur les effectifs n_{ij} en incluant un coefficient constant μ, un effet du `Groupe` α_i, un effet de l'intention β_j et une interaction γ_{ij}. Remarquer que ce modèle est sur-paramétré et qu'il est équivalent à celui incluant un coefficient constant μ, un effet `Groupe` α_i et une interaction γ_{ij}.
Ecrire la log-vraisemblance de cette dernière modélisation.
5) En posant $\tau_i = \sum_j \exp\big((\mu + \alpha_i) + \gamma_{ij}\big)$ trouver que la log-vraisemblance de ce modèle s'écrit

$$\mathcal{L}(n, \tau, \gamma) = \sum_i (n_i \log \tau_i - \tau_i) + \sum_i \sum_j n_{ij}\gamma_{ij} - n_i \log(\sum_j \exp(\gamma_{ij}))$$

Remarquons que la log-vraisemblance contient des termes en γ_{ij} d'un côté et des termes en τ_i de l'autre. La maximisation pour les termes en γ_{ij} ne dépend donc que des deux derniers termes et remarquer qu'ils sont égaux à la log-vraisemblance de la modélisation multinomiale de la variable `Intention` avec effet `Groupe`.
6) Ecrire une modélisation de Poisson sur les effectifs n_{ij} en incluant un coefficient constant μ, un effet `Groupe` α_i et un effet `Intention` β_j. On choisira la contrainte identifiante $\sum_{j=1}^{3} \beta_j = 0$ pour plus de simplicité. Ecrire ensuite la log-vraisemblance.
7) En posant $\tau_i = \sum_j \exp\big((\mu + \alpha_i) + \beta_j\big)$ montrer que cette log-vraisemblance s'écrit à une constante près

$$\mathcal{L}(Y, \tau, \beta) = \sum_i (n_i \log \tau_i - \tau_i) + \sum_i \sum_j n_{ij}\beta_j - n_i \log(\sum_j \exp(\beta_j))$$

et en déduire que la maximisation de la vraisemblance dans ce modèle de Poisson revient à celle de la modélisation multinomiale de la variable `Intention` sans effet `Groupe`.

Chapitre 13

Régularisation de la vraisemblance

Les procédures de sélection de variables présentées dans la partie III et dans la section 11.5 sont particulièrement utiles lorsque le nombre de variables explicatives est "grand". Elles permettent de diminuer le nombre de paramètres à estimer et donc de réduire la variance des estimateurs. Une alternative à cette stratégie est d'utiliser des techniques de régularisation. Ces techniques permettent en outre d'estimer les paramètres des GLM lorsque le nombre de variables p est plus grand que le nombre d'observations n. Ces approches ont été étudiées en détails dans le chapitre 8 pour le modèle de régression linéaire. Nous proposons ici de les étendre aux modèles GLM. Les idées sont identiques, nous allons contraindre l'espace des paramètres : la vraisemblance n'est plus maximisée sur $\mathbb{R}^p$ mais sur une boule de $\mathbb{R}^p$ centrée en 0.

13.1 Régressions ridge, lasso et elastic-net

On rappelle qu'un modèle GLM est défini par 3 composantes : une aléatoire, une déterministe et une de lien (voir définition 12.1). Jusqu'à présent nous écrivions la composante déterministe comme une combinaison linéaire des p variables explicatives et supposions que la première variable était égale à 1 si on souhaitait prendre en compte la constante dans le modèle. La constante jouant un rôle particulier dans les méthodes régularisées, il est important de la spécifier explicitement dans le modèle. C'est pourquoi, comme dans le chapitre 8, on utilisera la notation μ pour représenter la constante dans le modèle. On écrira alors le modèle logistique permettant d'expliquer une variable binaire Y par p variables explicatives

$$\mathsf{logit}(p_\beta(x)) = \mu + \beta_1 x_1 + \ldots + \beta_p x_p.$$

On considère un modèle GLM de log-vraisemblance $\mathcal{L}(Y, \mu, \beta)$. Nous avons vu dans les chapitres précédents que les approches standards d'estimation des paramètres

consistaient à chercher β dans $\mathbb{R}^p$ qui maximise cette log-vraisemblance (ou qui minimise son opposé). Les approches régularisées proposent de restreindre l'espace des paramètres. Par exemple, la régression ridge pose une contrainte sur la norme 2 des paramètres. Les estimateurs sont ainsi définis en minimisant l'opposé de la log-vraisemblance sous la contrainte que la norme 2 de β ne dépasse pas une valeur fixée. On cherche donc les valeurs de μ et de β qui minimisent

$$-\mathcal{L}(Y,\mu,\beta) \quad \text{sous la contrainte} \sum_{j=1}^{p} \beta_j^2 \leq \delta. \tag{13.1}$$

Le réel positif δ est ici un paramètre à calibrer. En effet, plus δ est petit, plus la contrainte est forte. Si on considère le cas extrême où $\delta = 0$, alors l'estimateur ridge est le vecteur nul. A l'inverse, la contrainte s'atténue lorsque δ augmente : l'estimateur ridge se rapproche de l'estimateur du maximum de vraisemblance pour des valeurs élevées de δ. En utilisant le Lagrangien, on a une écriture équivalente du problème d'optimisation [1] :

$$\hat{\beta}_{\text{ridge}}(\lambda) = \underset{\mu\in\mathbb{R},\beta\in\mathbb{R}^p}{\text{argmin}} \left[-\mathcal{L}(Y,\mu,\beta) + \lambda \sum_{j=1}^{p} \beta_j^2 \right] \tag{13.2}$$

avec $\lambda \geq 0$. L'équivalence est à prendre dans le sens où pour tout $\delta \geq 0$ il existe un unique $\lambda \geq 0$ (et réciproquement) tel que les solutions des problèmes d'optimisation (13.1) et (13.2) coïncident. On remarque que la constante μ n'apparaît pas dans la contrainte dans (13.1) ou dans la pénalité dans (13.2). L'approche lasso se base sur des contraintes de norme 1. On cherche μ et β qui minimisent

$$-\mathcal{L}(Y,\mu,\beta) \quad \text{sous la contrainte} \sum_{j=1}^{p} |\beta_j| \leq \delta \tag{13.3}$$

avec $\delta > 0$. Là encore, on a une écriture équivalente en passant au Lagrangien

$$\hat{\beta}_{\text{lasso}}(\lambda) = \underset{\mu\in\mathbb{R},\beta\in\mathbb{R}^p}{\text{argmin}} \left[-\mathcal{L}(Y,\mu,\beta) + \lambda \sum_{j=1}^{p} |\beta_j| \right] \tag{13.4}$$

avec $\lambda \geq 0$. Le paramètre λ contrôle le poids donné à la pénalité : lorsque $\lambda = 0$ on retrouve les estimateurs du maximum de vraisemblance alors qu'on obtient un estimateur égal à 0 lorsque λ tend vers $+\infty$. On obtiendra donc des estimateurs différents pour chaque valeur de λ.
Les méthodes ridge et lasso estiment les paramètres du modèle linéaire généralisé en maximisant la log-vraisemblance pénalisée par la norme 2 (ridge) et par la norme 1 (lasso). L'approche elastic-net (Zou & Hastie (2005)) propose de combiner ces deux pénalités. La pénalité elastic-net s'écrit donc

$$\lambda \sum_{j=1}^{p} ((1-\alpha)\beta_j^2 + \alpha|\beta_j|), \tag{13.5}$$

1. Par abus de notation $\hat{\beta}_{\text{ridge}}(\lambda)$ correspond $\hat{\mu}_{\text{ridge}}(\lambda)$ et $\hat{\beta}_{\text{ridge}}(\lambda)$

où $\alpha \in [0,1]$ et $\lambda \geq 0$.

$$\hat{\beta}_{\text{elast}}(\lambda) = \underset{\mu\in\mathbb{R},\beta\in\mathbb{R}^p}{\text{argmin}} \left[-\mathcal{L}(Y,\mu,\beta) + \lambda \sum_{j=1}^{p} ((1-\alpha)\beta_j^2 + \alpha|\beta_j|) \right]. \tag{13.6}$$

Le paramètre λ joue un rôle similaire à celui évoqué pour les méthodes ridge et lasso, il peut être sélectionné par les procédures présentées dans la section 13.2.
Le paramètre α représente le compromis entre les pénalités ridge et lasso : pour $\alpha = 0$, on retrouve l'estimateur ridge et on obtient l'estimateur lasso pour $\alpha = 1$. La méthode elastic-net est plus flexible que les approches ridge et lasso mais il faut sélectionner un paramètre supplémentaire : le paramètre α.
Les solutions des problèmes d'optimisation (13.2) et (13.4) s'obtiennent en utilisant des algorithmes numériques de programmation non linéaires (voir Hastie *et al.* (2001)). Le package **glmnet** utilise par exemple une méthode de descente par coordonnées pour calculer les estimateurs ridge, lasso et elastic-net de façon efficace. Nous le présentons à travers l'exemple des données `SAheart` :

```
> library(bestglm)
> data(SAheart)
```

Il n'est pas possible d'utiliser de formule dans la fonction **glmnet** pour spécifier la variable à expliquer et les variables explicatives. Il faut renseigner les variables explicatives dans une matrice. On utilise souvent la fonction **model.matrix** pour obtenir cette matrice :

```
> SAheart.X <- model.matrix(chd ~ ., data = SAheart)[,-1]
> SAheart.Y <- SAheart$chd
```

Si, parmi les variables explicatives, il y a des variables qualitatives, la fonction **model.matrix** fait un codage disjonctif des variables qualitatives. Ce codage remplace donc une variable admettant I modalités en I variables binaires. Pour chaque variable, la somme des variables issues du codage donne le vecteur constant dont toutes les valeurs valent 1. La fonction **model.matrix** supprime la colonne associée au codage de la première modalité afin qu'il n'y ait pas de problème de colinéarité.
Par défaut, la fonction **glmnet** choisit une grille de valeurs possibles pour λ et calcule les estimateurs pour chaque valeur de la grille. Le choix de la pénalité s'effectue à travers l'argument `alpha` de **glmnet**. On pourra par exemple calculer les estimateurs ridge, lasso et elastic-net avec les ordres suivants :

```
> library(glmnet)
> ridge <- glmnet(SAheart.X, SAheart.Y,family="binomial",alpha=0)
> lasso <- glmnet(SAheart.X, SAheart.Y,family="binomial",alpha=1)
> elast <- glmnet(SAheart.X, SAheart.Y,family="binomial",alpha=.5)
```

On représente souvent sur un graphique l'évolution des coefficients du modèle en fonction de λ ou de la norme 1 de $\hat{\beta}(\lambda)$. Un tel graphe est appelé *chemin de régularisation* ridge ou lasso ou elastic-net. On peut par exemple visualiser les chemins de régularisation ridge et lasso à l'aide des commandes suivantes (voir figure 13.1).

```
> plot(ridge, ylim=c(-0.1,0.2), main="Ridge")
> plot(ridge, ylim=c(-0.1,0.2), main="Ridge", xvar="lambda")
> plot(lasso, ylim=c(-0.1,0.2), main="Lasso")
> plot(lasso, ylim=c(-0.1,0.2), main="Lasso", xvar="lambda")
> plot(elast, ylim=c(-0.1,0.2), main="Elastic-net")
> plot(elast, ylim=c(-0.1,0.2), main="Elastic-net", xvar="lambda")
```

Les graphes de gauche représentent les valeurs des coefficients en fonction de la norme 1 du vecteur de coefficients tandis que ceux de droite montrent ces mêmes coefficients en fonction du logarithme de λ. Il y a bien entendu une « symétrie » entre ces deux représentations : une faible norme pour les paramètres correspond à une forte valeur de λ. On remarque que le lasso a tendance à mettre à 0 un certain nombre d'estimateurs.

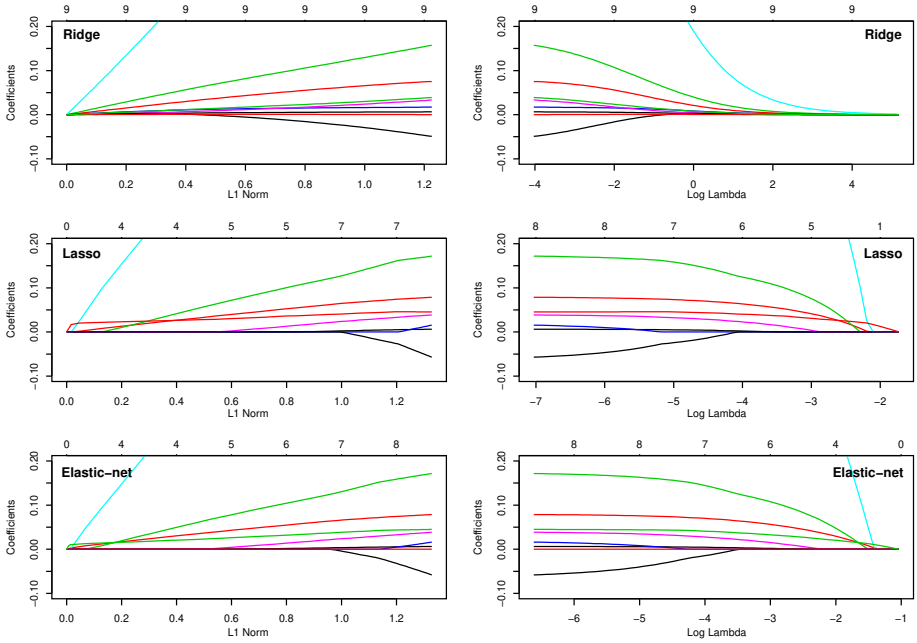


Fig. 13.1 – Chemins de régularisation ridge (haut), lasso (milieu) et elastic-net (bas) en fonction de $\|\hat{\beta}(\lambda)\|_1$ (gauche) et $\log(\lambda)$ (droite).

En effet, tous les coefficients sont à 0 lorsque λ est grand, ils « quittent 0 » les uns

après les autres au fur et à mesure que λ diminue. Cette approche permet donc de faire de la sélection de variables puisque pour une valeur de λ fixée, un certain nombre d'estimateurs seront égaux à 0. Les valeurs affichées en haut des graphes de la figure 13.1 indiquent le nombre de coefficients non nuls.

Remarque
Les variables sont généralement centrées réduites afin que la procédure ne dépende pas de l'échelle des valeurs prises par les variables. Il est possible de ne pas les standardiser en utilisant l'argument `standardize = FALSE` dans la fonction **glmnet**. Les valeurs de coefficients représentées sur la figure 13.1 sont les valeurs (re)calculées dans l'échelle des variables initiales. Ces valeurs se déduisent des valeurs obtenues sur les données centrées-réduites. On pourra se référer à la section 8.2 et à l'exercice 13.2 pour plus de détails.

13.2 Choix du paramètre de régularisation λ

Les approches ridge, lasso et elastic-net reviennent à minimiser

$$-\mathcal{L}(Y, \mu, \beta) + \lambda J(\beta) \tag{13.7}$$

ou encore

$$-\mathcal{L}(Y, \mu, \beta) \quad \text{sous la contrainte } J(\beta) \leq \delta$$

où $J(\beta)$ désigne la norme 2 pour la régression ridge, la norme 1 pour la régression lasso ou une combinaison des 2 pour elastic-net. Le choix du paramètre λ (ou δ) est crucial pour la performance de la procédure. Il va réguler le compromis biais-variance des estimateurs. Lorsque λ est grand, on augmente le poids de la pénalité dans le critère à optimiser. On va donc obtenir des estimateurs plus contraints avec moins de variance mais un biais qui risque d'être élevé (et réciproquement lorsque λ est petit). Il convient donc de trouver des procédures qui permettent de choisir ce paramètre de façon efficace.
Les méthodes classiques permettant de choisir le paramètre de régularisation λ consistent à :

1. donner un critère de performance ;
2. calculer le critère pour un ensemble de valeurs de λ ;
3. choisir la valeur de λ qui optimise l'estimation du critère.

Ces méthodes sont identiques à celles proposées au début de chapitre 10. Le critère se calcule par des méthodes de type validation croisée. Le choix du critère de performance dépend bien entendu du GLM considéré. Ce critère est le plus souvent basé sur une fonction de perte qui mesure l'erreur entre une prévision et une observation. On donne ci-dessous les fonctions de perte classiques pour les modèles linéaire, logistique et de Poisson.

Modèle linéaire.
La perte quadratique moyenne $\ell(y, m(x)) = (y - m(x))^2$ où

$$m(x) = \mu + \beta_1 x_1 + \ldots + \beta_p x_p = x'\beta.$$

Le choix de λ pour ce modèle a été présenté en détail dans la section 8.3.4.

Modèle de Poisson.
La déviance de la loi de Poisson

$$\ell(y, \lambda_\beta(x)) = 2(y \log(\lambda_\beta(x)) - \lambda_\beta(x))$$

où

$$\log(\lambda_\beta(x)) = \mu + \beta_1 x_1 + \ldots + \beta_p x_p = x'\beta.$$

On remarquera que la fonction $\lambda_\beta(x)$ représente l'espérance de la loi de Poisson et n'a rien à voir avec le paramètre de régularisation $\lambda \geq 0$ des méthodes pénalisées.

Modèle logistique.
On rappelle que ce modèle est défini par

$$\mathsf{logit}(p_\beta(x)) = \mu + \beta_1 x_1 + \ldots + \beta_p x_p = x'\beta.$$

On propose 3 critères qui sont implémentés dans la fonction **cv.glmnet**. Les critères erreur de classification et courbe ROC et aire sous la courbe (AUC) sont présentés en détail dans le chapitre suivant.

— La déviance binomiale

$$\ell(y, p_\beta(x)) = -2(y \log(p_\beta(x)) + (1 - y) \log(1 - p_\beta(x))). \tag{13.8}$$

C'est l'argument par défaut de la fonction.

— L'erreur de classification $\ell(y, g(x)) = \mathbb{1}_{g(x) \neq y}$ où

$$g_\beta(x) = \begin{cases} 1 & \text{si } p_\beta(x) \geq 0.5 \\ 0 & \text{sinon.} \end{cases}$$

C'est l'argument `class` du paramètre `type.measure`

— L'aire sous la courbe ROC (AUC, voir définition (14.1)) de la fonction de score $S(x) = p_\beta(x)$. Pour ce critère la fonction de perte se définit à partir de deux observations (x, y) et $(\tilde{x}, \tilde{y})$, une dans le groupe 1 ($y = 1$), l'autre dans le groupe 0 ($\tilde{y} = 0$) par (voir exercice 14.7)

$$\ell((y = 1, \tilde{y} = 0), (S_\beta(x), S_\beta(\tilde{x}))) = \mathbf{1}_{S_\beta(x) > S_\beta(\tilde{x})}.$$

C'est l'argument `AUC` du paramètre `type.measure`

Contrairement aux autres, ce critère est à maximiser mais, avec un léger abus de notation, nous conservons la terminologie fonction de perte et la notation ℓ.
Une fois la fonction de perte choisie, on choisit une grille qui contient différentes valeurs du paramètre de régularisation λ. On calcule ensuite, pour chaque valeur dans la grille, le critère de performance en utilisant des techniques d'apprentissage/validation ou validation croisée présentées dans les algorithmes 2 et 3 page 238. On choisira la valeur de λ qui optimise le critère choisi.
La fonction **cv.glmnet** du package **glmnet** utilise la validation croisée 10 blocs pour sélectionner le paramètre de régularisation. Le choix de la fonction de perte (et donc du critère) s'effectue à l'aide de l'option `type.measure`. On pourra par exemple choisir λ avec les critères de déviance, d'erreur de classification et d'AUC pour des régressions logistique ridge, lasso et elastic-net (avec $\alpha = 0.5$) avec :

```
> set.seed(2398)
> m1.ridge <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=0)
> m1.lasso <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=1)
> m1.elast <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=.5)
> m2.ridge <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=0,type.measure="class")
> m2.lasso <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=1,type.measure="class")
> m2.elast <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=.5,type.measure="class")
> m3.ridge <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=0,type.measure="auc")
> m3.lasso <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=1,type.measure="auc")
> m3.elast <- cv.glmnet(SAheart.X,SAheart.Y,family="binomial",
                 alpha=.5,type.measure="auc")
```

La fonction retourne, pour chaque valeur de λ testée :
- une erreur calculée par validation croisée (`cvm`) ainsi qu'une estimation de son écart-type (`cvsd`). On peut en déduire un intervalle de confiance (`cvlo` et `cvup`) associé à cette erreur ;
- le nombre de coefficients non nuls (`nzero`).

La valeur de λ qui minimise l'erreur (`lambda.min`) est également bien entendu proposée. La fonction renvoie de plus une autre valeur `lambda.1se` qui correspond à la plus grande valeur de λ pour laquelle l'erreur se situe à plus un écart type de l'erreur en `lambda.min`. En pratique, cela signifie que l'utilisateur peut choisir `lambda.min` ou `lambda.1se`. Si on privilégie la parcimonie du modèle (lorsqu'on fait du lasso par exemple), on choisira `lambda.1se`. On obtient ces deux valeurs de λ avec :

```
> m1.ridge$lambda.min
[1] 0.01774595
> m1.ridge$lambda.1se
[1] 0.2892148
```

Les différentes erreurs en fonction de $\log(\lambda)$ sont dessinées ci dessous (figure 8.6)

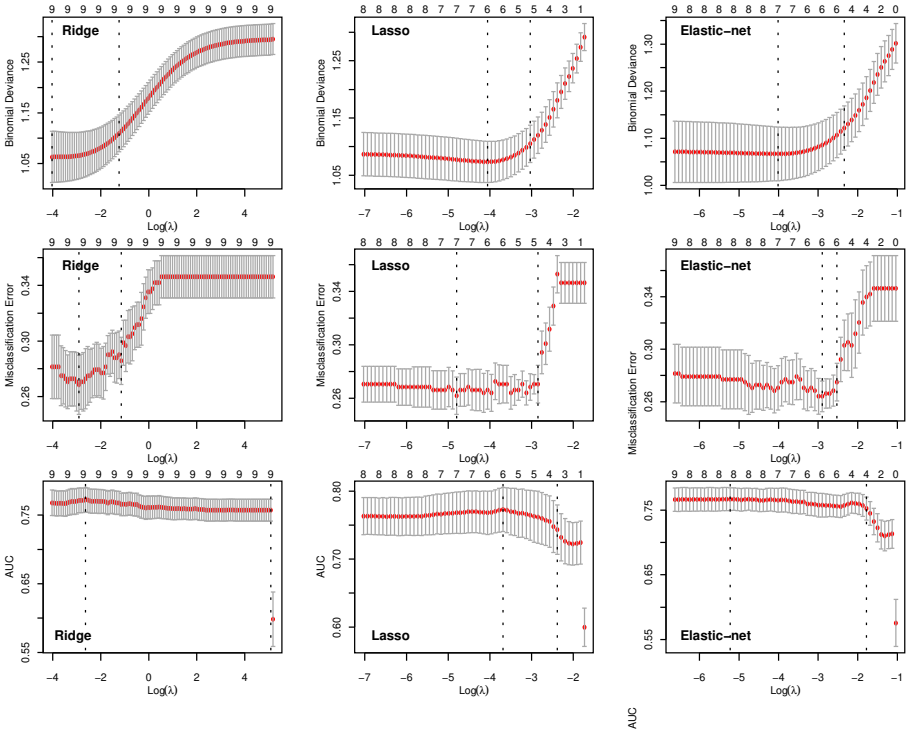


Fig. 13.2 – Déviance (haut), erreur de classification (milieu) et AUC (bas) pour les estimateurs ridge (gauche) et lasso (droite).

et obtenues avec les commandes suivantes :

```
> plot(m1.ridge,main="Ridge")
> plot(m1.lasso,main="Lasso")
> plot(m1.elast,main="Elastic-net")
> plot(m2.ridge,main="Ridge")
> plot(m2.lasso,main="Lasso")
> plot(m2.elast,main="Elastic-net")
> plot(m3.ridge,main="Ridge")
> plot(m3.lasso,main="Lasso")
> plot(m3.elast,main="Elastic-net")
```

On remarque la présence de deux lignes verticales sur chaque graphe. Celle de

gauche correspond à la valeur `lambda.min`, celle de droite à `lambda.1se`. On peut utiliser les sorties de **cv.glmnet** pour prédire. En plus de faire la validation croisée, cette fonction réajuste le modèle sur toutes les données pour prédire de nouveaux individus. On obtiendra par exemple les prévisions des probabilités de l'évènement `{chd=1}` pour les 15e et 50e individus du jeu de données avec les valeurs `lambda.min` et `lambda.1se` en appliquant la fonction **predict** à l'objet obtenu avec **cv.glmnet** :

```
> predict(m1.ridge,newx=SAheart.X[c(15,50),],
+         s="lambda.min",type="response")
          1
15 0.4896072
50 0.1326737
> predict(m1.ridge,newx=SAheart.X[c(15,50),],
+         s="lambda.1se",type="response")
          1
15 0.3893436
50 0.2479779
```

13.3 Group-lasso

Certaines applications nécessitent un traitement des variables explicatives par groupes. C'est par exemple le cas lorsqu'on dispose de variables explicatives qualitatives dans le modèle logistique. En présence de groupes de variables, il est souvent préférable d'annuler (ou pas) simultanément tous les coefficients d'un même groupe. Considérons par exemple le cas où on dispose de 3 variables explicatives X_1, X_2, X_3. X_1 et X_2 sont qualitatives et prennent respectivement pour valeurs A, B, C et D, E, F, G, X_3 est quantitative. La cible Y prend pour valeurs 0 ou 1. Le modèle logistique s'écrit

$$\begin{aligned}\mathsf{logit}\, p_\beta(x) =& \mu + \beta_1 \mathbf{1}_A(x_1) + \beta_2 \mathbf{1}_B(x_1) + \beta_3 \mathbf{1}_C(x_1) \\ &+ \beta_4 \mathbf{1}_D(x_2) + \beta_5 \mathbf{1}_E(x_2) + \beta_6 \mathbf{1}_F(x_2) + \beta_7 \mathbf{1}_G(x_2) + \beta_8 x_3. \end{aligned} \tag{13.9}$$

muni des contraintes $\beta_1 = \beta_4 = 0$. L'approche lasso appliquée à cet exemple n'est pas forcément satisfaisante : elle risque d'annuler certains coefficients β_j sans prendre en compte le fait que les variables X_1 et X_2 sont représentées par les groupes de coefficients (β_2, β_3) et $(\beta_5, \beta_6, \beta_7)$. Le group-lasso permet de pallier ce problème en traitant les variables par groupes.

Considérons donc le cas où les p variables explicatives $X_1, \ldots, X_p$ sont regroupées en L groupes $\mathbf{X}_1, \ldots, \mathbf{X}_L$ de cardinaux $p_1, \ldots, p_L$ tels que $p_1 + \ldots + p_L = p$. Sans perte de généralité, nous supposerons que les variables sont ordonnées en fonction des groupes, c'est-à-dire $\mathbf{X}_1 = \{X_1, \ldots, X_{p_1}\}$, ... On note $\tilde{\beta}_\ell, \ell = 1, \ldots, L$ le vecteur

des coefficients associé au groupe $\mathbf{X}_\ell$. Le modèle logistique s'écrit donc

$$\begin{aligned}\mathsf{logit}\, p_\beta(x) =& \mu + \beta_1 x_1 + \ldots + \beta_p x_p \\ =& \mu + \tilde{\beta}_1' \mathbf{x}_1 + \ldots + \tilde{\beta}_\ell' \mathbf{x}_\ell\end{aligned}$$

Les *estimateurs group-lasso* s'obtiennent en minimisant le critère

$$-\mathcal{L}(Y, \mu, \beta) + \lambda \sum_{\ell=1}^{L} \sqrt{p_\ell} \|\tilde{\beta}_\ell\|_2.$$

Seule la pénalité change par rapport à la méthode lasso, elle s'écrit comme la somme des normes des groupes de coefficients pondérée par la taille des groupes. Cette pénalité est proche de la pénalité lasso. Le lasso s'écrit comme une combinaison linéaire des valeurs absolues de paramètres tandis que le group-lasso utilise une combinaison linéaire de normes de paramètres. Ainsi les solutions du problème ci-dessus auront tendance à vérifier $\|\tilde{\beta}_\ell\|_2 = 0$ pour certains groupes et donc à mettre à 0 tous les coefficients de ces groupes. Pour finir, remarquons que le coefficient $\sqrt{p_\ell}$ dans la pénalité group-lasso permet de prendre en compte la taille des groupes : la pénalité est plus importante pour les grands groupes.

Exemple 13.1
On considère le modèle (13.9) pour le jeu de données suivant :

```
> X1 <- c(rep("A",60), rep("B",90), rep("C",50))
> X2 <- c(rep("D",40), rep("E",60), rep("F",55), rep("G",45))
> set.seed(23)
> X3 <- runif(200)
> Y <- 2*round(runif(200)) - 1
> donnees <- data.frame(X1, X2, X3, Y)
```

On compare les procédures lasso et group-lasso en regroupant les coefficients selon $\tilde{\beta}_1 = (\beta_2, \beta_3)$, $\tilde{\beta}_2 = (\beta_5, \beta_6, \beta_7)$ et $\tilde{\beta}_3 = \beta_8$. Le package **gglasso** permet de faire du group-lasso.

```
> D <- model.matrix(Y~.,data=donnees)[,-1]
> lasso <- glmnet(D,Y, alpha=1, lambda=exp(seq(-3,-5,length=100)))
> groupe <- c(1, 1, 2, 2, 2, 3)
> library(gglasso)
> Y1 <- 2*Y-1
> g.lasso <- gglasso(D, Y1, group=groupe, loss="logit",
                     lambda=exp(seq(-4, -5.5, length=100)))
```

On obtient les chemins de régularisation lasso et group-lasso avec les commandes suivantes (voir figure 13.3).

```
> plot(lasso,xvar="lambda",lwd=2,main="Lasso")
> plot(g.lasso,main="Group-lasso")
```

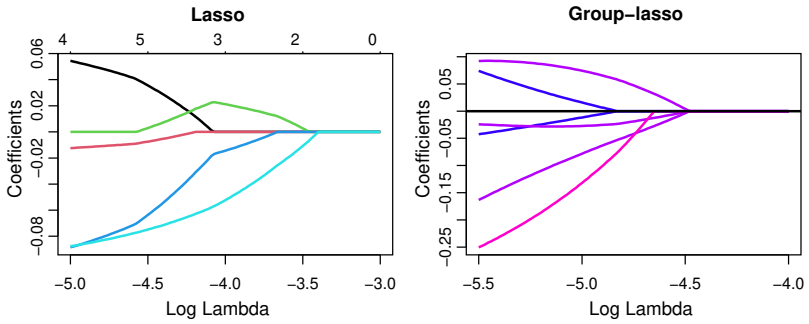


Fig. 13.3 – Chemins de régularisation lasso (gauche) et group-lasso (droite).

On voit clairement que, lorsque λ augmente, les coefficients s'annulent de façon séquentielle les uns après les autres pour l'approche lasso tandis qu'ils s'annulent par groupes pour le group-lasso : d'abord les 2 coefficients associés à X_1, puis le coefficient associé à X_3 et enfin les 3 associés à X_2.

Comme pour les méthodes ridge et lasso, il est important de bien sélectionner le paramètre de régularisation λ pour le group-lasso. La méthode est identique, on calcule une erreur par validation croisée pour une grille de valeurs de λ et on choisit la valeur qui minime l'erreur. On pourra utiliser la fonction **cv.gglasso** de **gglasso**.

13.4 Exercices

Exercice 13.1 (Questions de cours)

1) $\lambda \geq 0$, les estimateurs ridge s'obtiennent en pénalisant l'opposé de la vraisemblance par

A. $\lambda \sum_{j=1}^{p} |\beta_j|$,
B. $\lambda \sum_{j=1}^{p} \sqrt{\beta_j}$,
C. $\lambda \sum_{j=1}^{p} \beta_j^2$,
D. $\lambda \sum_{j=1}^{p} \log(\beta_j^2)$,
E. $\lambda \sum_{j=1}^{p} \log(|\beta_j|)$,
F. $\lambda \sum_{j=1}^{p} \beta_j$.

2) $\lambda \geq 0$, les estimateurs lasso s'obtiennent en pénalisant l'opposé de la vraisemblance par

A. $\lambda \sum_{j=1}^{p} |\beta_j|$,
B. $\lambda \sum_{j=1}^{p} \sqrt{\beta_j}$,
C. $\lambda \sum_{j=1}^{p} \beta_j^2$,
D. $\lambda \sum_{j=1}^{p} \log(\beta_j^2)$,
E. $\lambda \sum_{j=1}^{p} \log(|\beta_j|)$,
F. $\lambda \sum_{j=1}^{p} \beta_j$.

3) Parmi les affirmations suivantes, lesquelles sont vraies ?
A. Les méthodes régularisées permettent de réduire la variance des estimateurs du MV,
B. Les méthodes régularisées permettent de réduire le biais des estimateurs du MV,
C. On utilise généralement les méthodes ridge/lasso lorsque le nombre de variables explicatives p est grand,
D. Les estimateurs ridge/lasso sont toujours plus performants que les estimateurs du MV.

4) On considère les estimateurs lasso définis par la pénalité proposée à la question précédente. Parmi les affirmations suivantes, lesquelles sont vraies ?
A. Les estimateurs seront proches de 0 pour de très petites valeurs de λ,
B. Les estimateurs seront proches des estimateurs du MV pour de très grandes de λ,
C. Les estimateurs seront proches de 0 pour de très grandes valeurs de λ,

D. Les estimateurs seront proches des estimateurs du MV pour de très petites de λ,
E. Il faut toujours choisir λ le plus grand possible,
F. Il faut toujours choisir λ le plus petit possible.

Exercice 13.2 (Lasso sur données centrées réduites)
On considère le modèle lasso obtenu sur les données `SAheart` à l'aide des commandes

```
> library(bestglm),data(SAheart)
> SAheart.X <- model.matrix(chd~.,data=SAheart)[,-1]
> SAheart.Y <- SAheart$chd
> mod.lasso <- glmnet(SAheart.X,SAheart.Y,family="binomial",alpha=1)
```

L'objet `mod.lasso` contient (entre autres) les valeurs des estimateurs lasso calculées pour chaque valeur de λ contenue dans le vecteur suivant

```
> lam.lasso <- mod.lasso$lambda
```

1) A l'aide de la fonction **coef** retrouver les estimateurs des paramètres du modèle correspondant à la 50e valeur du vecteur `lam.lasso`.
2) Centrer réduire la matrice `SAheart.X` et calculer les estimateurs lasso sur les données centrées-réduites.
3) Retrouver les valeurs obtenues question 1) à partir des estimateurs calculées question 2.

Exercice 13.3 (Comparaison de méthodes et courbes ROC)
Le fichier `logit_ridge_lasso.csv` contient $n = 500$ observations de 100 variables quantitatives $X_1, \ldots, X_{100}$ et d'une variable binaire Y. Le problème est d'expliquer Y par $X_1, \ldots, X_{100}$.
1) Séparer les données en un échantillon d'apprentissage de taille 200 et un échantillon test de taille 300.
2) A partir des donnés d'apprentissage **uniquement**, construire sur R :
   a) le modèle logistique complet (celui avec les 100 variables explicatives),
   b) un modèle logistique utilisant une procédure de descendante (fonction **step**),
   c) un modèle logistique lasso et ridge utilisant la déviance pour sélectionner λ,
   d) un modèle logistique lasso et ridge utilisant l'AUC pour sélectionner λ.
3) Représenter les courbes ROC des 6 modèles construits calculées sur l'échantillon test.
4) Calculer et comparer les AUC de ces 6 modèles.

Exercice 13.4 (Surapprentissage)
1) Pour les 6 modèles construits dans l'exercice précédent, estimer les probabilités de l'évènement $\{Y = 1\}$ pour les individus de l'échantillon d'apprentissage.
2) En déduire une règle de prévision. Pour la règle choisie, comparer les labels prédits par les 6 modèles aux labels observés (toujours pour l'échantillon d'apprentissage).
3) Reprendre les deux questions précédentes sur les individus de l'échantillon test. Interpréter.

Chapitre 14

Comparaison des différentes méthodes en classification supervisée, études de cas réel

Dans le chapitre 10, nous avons comparé différentes méthodes de régression en analysant les erreurs de prévisions obtenues par validation croisée. Ici, nous souhaitons faire de même avec des méthodes de classification. La variable Y à prédire n'est plus une variable quantitative mais qualitative à deux modalités, ce qui va induire quelques différences, tant au niveau de la prévision que des critères de qualité de celle-ci. Commençons par présenter la prévision en classification supervisée.

14.1 Prévision en classification supervisée

Considérons (X, Y) un couple aléatoire à valeurs dans $\mathbb{R}^p \times \{0, 1\}$. Le problème consiste à expliquer, ou à prévoir, le label Y par X. Il faut donc choisir une règle de classification ou de prévision notée g :

$$\begin{aligned} g : \mathbb{R}^p &\to \{0, 1\} \\ x &\mapsto g(x) \end{aligned}$$

qui à un individu dont les variables valent $x \in \mathbb{R}^p$ associe son label $g(x) \in \{0, 1\}$. Une manière naturelle d'obtenir une règle est d'analyser la probabilité $p_\beta(x) = \Pr_\beta(Y = 1|X = x)$. Plus cette probabilité est élevée, plus il semble évident de prédire le groupe 1. Le passage de cette probabilité à la règle de prévision s'effectue en choisissant un seuil $s \in \mathbb{R}$: si la probabilité est plus petite que s, on décidera que le label vaut 0 et si la probabilité estimée est plus grande que s, on décidera que le label vaut 1 :

$$g_s(x) = \begin{cases} 1 & \text{si } p_\beta(x) \geq s \\ 0 & \text{sinon.} \end{cases}$$

Bien entendu le seuil s est inconnu et en pratique il doit être choisi par l'utilisateur. Un choix classique consiste à prendre $s = 0.5$. Bien que ce choix possède certaines justifications théoriques (exercice 14.2, p. 351), rien ne dit qu'il soit optimal et on évoquera dans la suite différentes manières de le sélectionner (voir section 14.4).
Outre le problème du choix du seuil, la probabilité $p_\beta(x)$ est inconnue et on doit l'estimer à partir d'un échantillon $(X_1, Y_1), \ldots, (X_n, Y_n)$ de même loi que (X, Y). Nous avons fait cela dans les chapitres précédents pour le modèle logistique avec les estimations par maximum de vraisemblance, ridge, lasso... Une fois l'estimateur $p_{\hat\beta}(x)$ obtenu, nous déduirons l'estimateur $\hat g_s$ de g_s en remplaçant $p_\beta(x)$ par $p_{\hat\beta}(x)$.
Nous remarquons ainsi que la classification binaire fait intervenir deux quantités fortement liées : la règle de prévision (estimée) $\hat g_s(x)$ et la probabilité (estimée) $p_{\hat\beta}(x)$. La probabilité $p_{\hat\beta}(x)$ donne plus d'information que le label prédit. Ainsi pour un seuil choisi à $s = 0.5$, on prévoit le même label $\hat g_s(x) = 1$ pour $p_{\hat\beta}(x) = 0.501$ ou $p_{\hat\beta}(x) = 0.999$. Plus la probabilité est forte (ou faible), plus on sera confiant dans le label associé. La probabilité permet également d'ordonner les valeurs : du plus probablement 0 vers le plus probablement 1 en passant par le milieu plus incertain. Cet ordre, impossible à obtenir avec seulement les labels, est à la base même du concept de scoring.
Construire un score revient simplement à trouver une procédure qui permette de donner une note à des individus qui soit en lien avec Y dans le sens où :

— une note élevée signifie que l'individu a de grandes chances (ou une grande probabilité) d'être dans le groupe 1 ;
— une note faible signifie que l'individu a de grandes chances (ou une grande probabilité) d'être dans le groupe 0.

Mathématiquement, un score est donc simplement une fonction $S : \mathbb{R}^p \to \mathbb{R}$ qui, à un individu $x \in \mathbb{R}^p$, associe la note $S(x) \in \mathbb{R}$. Dans notre contexte, une façon naturelle de noter un individu est de considérer la probabilité qu'il soit dans le groupe 1, c'est-à-dire de considérer $S(x) = p_\beta(x)$ (ou plus exactement son estimation $\hat S(x) = p_{\hat\beta}(x)$). C'est ce score que nous considérerons dans ce chapitre.

Remarque : la valeur de la note de score $S(x)$ n'est généralement pas importante en elle-même. C'est l'ordre dans lequel un score va classer des individus qui compte. Ainsi, on dira que si $\phi : \mathbb{R} \to \mathbb{R}$ est une fonction bijective croissante, alors les scores S et $\phi \circ S$ sont équivalents.

Pour résumer, la prévision en classification binaire s'effectue à travers deux quantités : la probabilité (ou le score) et la règle. Et donc si l'on doit comparer plusieurs algorithmes de prévision, par exemple régression logistique, ridge, lasso, il est possible de les comparer via les probabilités (ou scores) ou via les règles associées (à condition d'avoir choisi un seuil s). Ces comparaisons s'effectueront à l'aide de critères de performance spécifiques. Pour un score, on évaluera sa performance à travers sa capacité à bien ordonner des individus, ce qui revient à mesurer si les individus du groupe 0 sont bien plus faiblement notés que ceux du groupe 1. Pour une règle de classification, on cherchera à évaluer si la règle prédit bien de façon globale, ou si elle est capable de bien identifier les positifs (groupe 1) ou négatifs (groupe 0). Nous présentons ces critères dans les parties suivantes.

14.2 Performance d'une règle

En classification binaire, nous avons un modèle théorique où $\mathbb{E}(Y|X = x) = p(x)$. La règle associée est définie par $g_s(x) = \mathbb{1}_{p(x)>s}$ et elle permet de déduire le label de Y en fonction des valeurs des variables explicatives x et du seuil s. Y prend deux valeurs possibles, 0 ou 1, et la règle de classification (au seuil donné s) prend aussi deux valeurs, 0 ou 1. Ces cas sont résumés dans la table de confusion théorique (tableau 14.1) où sont quantifiées les probabilités.

Vrai \ Règle	$g_s(X) = 0$	$g_s(X) = 1$
$Y = 0$	$\Pr(Y = 0 \cap g_s(X) = 0)$	$\Pr(Y = 0 \cap g_s(X) = 1)$
$Y = 1$	$\Pr(Y = 1 \cap g_s(X) = 0)$	$\Pr(Y = 1 \cap g_s(X) = 1)$

Tableau 14.1 – Table de confusion pour une règle théorique.

Même si la règle g_s est connue, le label Y est une variable aléatoire qui suit ici une loi de Bernoulli et il est donc impossible [1] de trouver exactement le label Y à partir de X avec une règle. Autrement dit, si le modèle générant les données est une Bernoulli de paramètre $p(x)$ connu, on sait seulement que $\mathbb{E}(Y|X = x) = p(x)$ mais on ne peut pas connaître les valeurs de Y.
A partir de l'échantillon $(X_1, Y_1), \ldots, (X_n, Y_n)$, échantillon noté aussi $\mathcal{D}_n$, nous avons estimé la règle théorique par $\hat{g}_s$. En appliquant cette règle estimée, nous obtenons des estimations du label. Il faut ensuite évaluer/estimer les probabilités inconnues de la table de confusion 14.1. Une manière naturelle d'estimer une probabilité est de calculer une fréquence et nous comptons donc les effectifs de chaque cellule. On obtient ainsi la matrice de confusion empirique (tableau 14.2) qui est une estimation des probabilités multipliées par le nombre d'individus.

Vrai \ Prévu	$\hat{g}_s(X) = 0$	$\hat{g}_s(X) = 1$
$Y = 0$	Nombre VN de Vrais Négatifs	Nombre FP de Faux Positifs
$Y = 1$	Nombre de FN Faux Négatifs	Nombre de VP Vrais Positifs

Tableau 14.2 – Matrice de confusion empirique pour une règle estimée.

Dans cette matrice nous avons introduit une nouvelle terminologie à base de négatifs/positifs. Celle-ci provient d'un cas très répandu en médecine : la découverte de la présence d'une maladie. Dans ce cas-là, il est d'usage d'affecter à un patient malade le label $Y = 1$ et on cherche à détecter la présence de la maladie. Si la probabilité estimée est inférieure à s, le label prévu est alors 0 et on parle alors de test négatif. Si par contre la probabilité estimée est supérieure à s, le label prévu est alors 1 et on parle de test positif.

1. Sauf cas de séparation avec $p(x) = 0$ ou 1.

La question qui se pose est comment calculer cette matrice de confusion.

Calcul à partir des données initiales
Ici, les n données initiales de l'échantillon $\mathcal{D}_n$ sont utilisées pour calculer les valeurs de la matrice de confusion 14.2. On utilise donc l'échantillon $\mathcal{D}_n$ deux fois : une première fois pour estimer les paramètres du modèle et obtenir la règle estimée $\hat{g}_s$, et une deuxième fois pour calculer les prévisions $\hat{g}_s(X_i)$; on parle ici d'ajustement. Ces ajustements sont enfin comparés aux observations Y_i. Nous avons alors une table de confusion ajustée. Nous avons déjà évoqué la différence entre erreur d'ajustement et erreur de prévision. Les résultats calculés en ajustement seront toujours meilleurs que des résultats calculés sur de nouvelles données. Imaginons un modèle qui sur-apprend, il fera donc peu d'erreur et la table de confusion aura alors peu de valeurs sur les termes hors diagonale. Cependant ce modèle sera trop spécifique aux données et il donnera sûrement de mauvaises prévisions. Nous ne préconisons donc pas l'utilisation de cette table de confusion ajustée.

Calcul à partir d'apprentissage/validation
A partir des n données initiales de l'échantillon $\mathcal{D}_n$, on propose une séparation des données en deux parties : une partie dédiée à l'apprentissage, une autre à la validation. On construit (on estime) la règle $\hat{g}_s$ sur l'échantillon d'apprentissage et on applique cette règle sur l'échantillon de validation en utilisant la règle $\hat{g}_s$. On compare les labels prévus de l'échantillon de validation aux labels observés. Nous aurons alors une table de confusion prévue. Lors du découpage, on privilégie en général l'échantillon d'apprentissage qui comporte alors trois quarts des observations en général mais nous recommandons la procédure suivante.

Calcul à partir d'une validation croisée
On découpe l'échantillon initial en K blocs, blocs généralement tirés au hasard. Ensuite, la procédure apprentissage/validation est répétée K fois en considérant chaque bloc comme échantillon de validation, les $K-1$ blocs restants constituent l'échantillon d'apprentissage. Schématisons cette procédure pour une validation croisée $K = 4$ blocs, la figure 14.1 représente l'étape de séparation initiale :

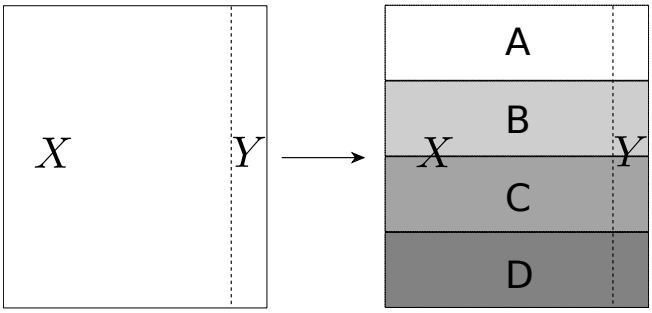


Fig. 14.1 – Validation croisée K-blocs ($K = 4$) : séparation en blocs.

Une fois les blocs constitués, nous commençons par utiliser le bloc A en validation, les autres en apprentissage (voir fig. 14.2). Les blocs B, C et D sont donc utilisés pour estimer la règle $\hat{g}_s$ que l'on utilise pour prédire la probabilité ou le label des individus du bloc A. On note $\hat{P}$ le vecteur qui contient ces prévisions, cela peut être les probabilités ou directement le label. Dans la pratique c'est souvent les probabilités qui sont conservées.

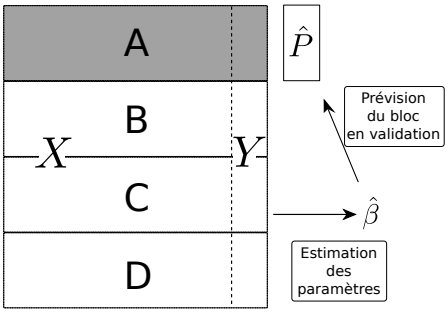


Fig. 14.2 – Validation croisée K blocs (avec $K = 4$, pour un modèle de régression) : seconde étape, le bloc A est utilisé en validation, les autres en apprentissage.

On répète ensuite cette procédure sur les blocs restants : on considère tour à tour les blocs B, C et D comme échantillon de validation afin d'obtenir une probabilité estimée pour tous les individus (voir fig. 14.3).

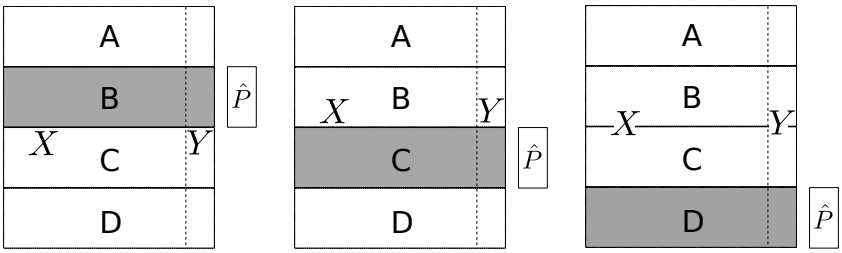


Fig. 14.3 – Validation croisée K blocs (avec $K = 4$) : étapes 3, 4 et 5, les blocs B, C et D sont respectivement utilisés en validation, les autres en apprentissage.

A la fin de la procédure, nous avons pour toutes les observations le couple Y mesuré et $\hat{P}$ sa probabilité (ou son label) prévue. Si nous avons une probabilité, nous fixons un seuil et calculons la table de confusion.
L'algorithme a été illustré pour $K = 4$ mais en pratique on utilise souvent $K = 10$. Lorsque $K = n$, on parle de validation croisée *leave-one-out* (l'abréviation "loo" est fréquemment utilisée).

Remarques

— Les algorithmes sont ici présentés pour une méthode de prévision. Si cette méthode nécessite un découpage des données, il faudra découper uniquement

l'échantillon d'apprentissage (ou l'échantillon privé du k^{e} bloc pour la validation croisée). Si par exemple on souhaite faire de la régression lasso en sélectionnant le paramètre de régularisation par validation croisée, alors seul l'échantillon d'apprentissage (ou l'échantillon privé du k^{e} bloc) devra être découpé en bloc pour la sélection du paramètre.

— Lorsque la méthode de prévision dépend d'un paramètre, la valeur de ce paramètre ne sera pas forcément la même à chaque itération de la validation croisée. Par exemple, si on utilise une méthode de sélection pas à pas (voir section 7.4.2), les variables sélectionnées à chaque étape de la validation croisée ne seront pas forcément les mêmes. Les algorithmes 2 et 3 sont utilisés pour choisir une procédure (régression linéaire avec toutes les variables ou avec sélection de variable ou régression ridge/lasso...). Une fois la procédure choisie, on pourra la réajuster sur toutes les données pour obtenir le modèle final.

— Les algorithmes sont ici présentés avec une règle de classification (donc avec un seuil s choisi) mais il est également possible de comparer les méthodes sans se fixer de seuil s en utilisant par exemple les courbes ROC (et les AUC) des différentes méthodes. On choisira ensuite la méthode ayant l'AUC le plus élevé et on cherchera ensuite le seuil s.

En conclusion, en fonction du type de procédure utilisé pour évaluer la matrice de confusion empirique, les résultats seront en général bien différents. Les critères estimés à partir d'une matrice de confusion évaluée par ajustement apparaîtront en général meilleurs que les mêmes critères estimés à partir d'une matrice de confusion évaluée par apprentissage/validation, apparence qui pourrait être trompeuse. Nous conseillons donc fortement d'évaluer les critères sur des données qui n'ont pas servi à l'estimation ou dit autrement d'utiliser une matrice de confusion empirique évaluée par apprentissage/validation ou validation croisée si les effectifs sont trop faibles.

Enfin, rappelons que la table de confusion est définie à partir d'une règle qui dépend d'un seuil s : les valeurs de la table $\text{FP}, \text{FN}, \text{VP}, \text{VN}$ varient donc avec s ainsi que toutes les valeurs qui en découlent. Nous souhaitons ici bien marquer que les critères dépendent de s, mais, afin de ne pas alourdir les notations, les valeurs de la table $(\text{FP}, \text{FN}, \text{VP}, \text{VN})$ ne seront pas écrites comme fonction de s même si en toute rigueur nous le devrions.

14.2.1 Erreur de classification et *accuracy*

Le premier critère d'évaluation d'une règle de classification g_s est l'erreur de classification. Elle est définie par $E_c(s) = P(g_s(X) \neq Y)$ et peut être estimée par

$$\hat{E}_c(s) = \frac{\text{FP} + \text{FN}}{\text{FP} + \text{FN} + \text{VP} + \text{VN}}.$$

L'*accuracy* (ou précision) est définie comme la probabilité d'être bien classé $A_c(s) = P(g_s(X) = Y)$ et correspond donc à $1 - E_c(s)$ et elle peut être estimée par

$$\hat{A}_c(s) = \frac{\text{VP} + \text{FN}}{\text{FP} + \text{FN} + \text{VP} + \text{VN}}.$$

Ces deux critères sont indépendants du choix des labels $Y = 0$ ou $Y = 1$ puisque l'on voit qu'ils jouent les mêmes rôles.

14.2.2 Sensibilité (*recall*) et taux de faux négatifs

La sensibilité (ou *recall* ou rappel) est définie par $S_e(s) = P(g_s(X) = 1|Y = 1)$, cela correspond à la probabilité que la règle détecte les positifs parmi les positifs. La sensibilité peut être estimée par

$$\hat{S}_e(s) = \frac{\mathrm{VP}}{\mathrm{VP} + \mathrm{FN}}.$$

Une sensibilité égale à 1 correspond à une règle qui a trouvé tous les positifs. Mais trouver tous les positifs est simple en proposant par exemple une règle qui indique que tout le monde est positif, ce que l'on peut faire en proposant un seuil s qui vaut 0.
Une autre grandeur liée à la sensibilité est le taux de faux négatifs (*False Negative Rate* en anglais) :

$$\beta(s) = \mathrm{FNR} = \Pr(g_s(X) = 0|Y = 1)$$

qui est simplement $1 - S_e(s)$. Ce taux $\beta(s)$ est à rapprocher de l'erreur de première espèce en théorie des tests. Son estimation est $1 - \hat{S}_e(s)$, ce qui vaut

$$\hat{\beta}(s) = \frac{\mathrm{FN}}{\mathrm{VP} + \mathrm{FN}}.$$

Ces deux grandeurs ne sont pas indépendantes du choix des labels $Y = 0$ ou $Y = 1$. L'utilisateur, avant de les interpréter ou de les utiliser, devra donc choisir ses labels et adapter son interprétation aux labels choisis.

14.2.3 Spécificité et taux de faux positifs

La spécificité est définie par $S_p(s) = P(g_s(X) = 0|Y = 0)$, cela correspond à la probabilité que la règle détecte les négatifs parmi les négatifs. La spécificité peut être estimée par

$$\hat{S}_p(s) = \frac{\mathrm{VN}}{\mathrm{VN} + \mathrm{FP}}.$$

Une spécificité égale à 1 correspond à une règle qui a trouvé tous les négatifs. Mais trouver tous les négatifs est simple en proposant par exemple une règle qui indique que tout le monde est négatif ce que l'on peut faire en proposant un seuil s qui vaut 1.
Une autre grandeur liée à la spécificité est le taux de faux positifs
(*False Positive Rate* en anglais) :

$$\alpha(s) = \mathrm{FPR} = \Pr(g_s(X) = 1|Y = 0)$$

qui est simplement $1 - S_p(s)$. Ce taux $\alpha(s)$ est à rapprocher de l'erreur de seconde espèce en théorie des tests. Son estimation est $1 - \hat{S}_p(s)$, ce qui vaut

$$\hat{\alpha}(s) = \frac{\mathrm{FP}}{\mathrm{VN} + \mathrm{FP}}.$$

Ces deux grandeurs ne sont pas indépendantes du choix des labels $Y = 0$ ou $Y = 1$. L'utilisateur, avant de les interpréter ou de les utiliser, devra donc choisir ses labels et adapter son interprétation aux labels choisis.

14.2.4 Mesure sur les tables de contingence

Remarquons que la matrice de confusion empirique est une table de contingence 2×2 sur laquelle nous pouvons utiliser des mesures standards d'association que nous rappelons ci-après.
La plus classique d'entre elles est le χ^2. Le nombre de mesures dans la matrice de confusion empirique $n^\star$ est considéré comme déterminé par l'utilisateur (non aléatoire). Le χ^2 mesure l'écart au carré entre les effectifs observés et les effectifs théoriques sous hypothèse d'indépendance, cet écart étant divisé par les effectifs théoriques :

$$\chi^2(s) = \sum_{i=0}^{1} \sum_{j=0}^{1} \frac{\big(n^\star \Pr(Y = i \cap g_s(X) = j) - n^\star \Pr(Y = i) \Pr(g_s(X) = j)\big)^2}{n^\star \Pr(Y = i) \Pr(g_s(X) = j)}.$$

Cette mesure théorique du χ^2 est estimée ici et donne (en remplaçant les probabilités inconnues par les fréquences correspondantes et après de nombreuses simplifications) :

$$\chi^2(s) = \frac{n^\star(\mathrm{VP}\,.\,\mathrm{VN} - \mathrm{FN}\,.\,\mathrm{FP})^2}{(\mathrm{VN} + \mathrm{FP})(\mathrm{FN} + \mathrm{VP})(\mathrm{VN} + \mathrm{FN})(\mathrm{FP} + VP)}.$$

Nous voyons qu'il s'agit d'une grandeur proportionnelle à l'écart au carré entre valeurs biens classées et celles mal classées. Bien sûr, plus le χ^2 est élevé, meilleure est notre règle de classification. Le coefficient ϕ de Pearson est défini par

$$\phi(s) = \frac{\mathrm{VP}\,.\,\mathrm{VN} - \mathrm{FN}\,.\,\mathrm{FP}}{\sqrt{(\mathrm{VN} + \mathrm{FP})(\mathrm{FN} + \mathrm{VP})(\mathrm{VN} + \mathrm{FN})(\mathrm{FP} + VP)}}.$$

Il varie entre -1 et $+1$ et est relié au $\chi^2(s)$ ($\chi^2(s) = n^\star \phi^2(s)$). L'interprétation de ce coefficient est simple : 1 classification parfaite, 0 pas de lien et -1 lien opposé (la classification donne 0 pour $Y = 1$ et 1 pour $Y = 0$, ce qui est à rapprocher plutôt d'une erreur dans les programmes...) Dans les deux cas, ces mesures sont indépendantes du choix du codage 0 ou 1 pour Y.
Tous les critères que nous venons de présenter sont définis à partir d'une règle $\hat{g}_s$ qui nécessite de fixer un seuil. Bien évidemment si le seuil s évolue, tous les critères associés vont évoluer. Quelle est l'évolution avec s de la qualité de classification et serait-il possible de penser à un critère global (pour un ensemble de s) ? Il existe quelques critères qui ne dépendent pas du seuil s et donc pour être cohérent avec nos notations, nous les définirons pour le score généralisant la règle pour tout s.

14.3 Performance d'un score

14.3.1 Courbe ROC

L'idée de la courbe ROC (de l'anglais *Receiver Operating Characteristic*) est de proposer une représentation graphique de l'évolution de la qualité de classification avec la valeur du seuil s. La courbe ROC permet de visualiser simultanément les deux types d'erreur (ou par symétrie les spécificité et sensibilité) pour toutes les valeurs de seuil s possibles.

Définition 14.1
La courbe ROC d'un score $S : \mathbb{R}^p \to \mathbb{R}$ est une courbe paramétrée par le seuil $s \in \mathbb{R}$ d'abscisse

$$x(s) = \alpha(s) = 1 - S_p(s) = \Pr(S(X) > s | Y = 0)$$

et d'ordonnée

$$y(s) = 1 - \beta(s) = S_e(s) = \Pr(S(X) > s | Y = 1).$$

Dit autrement, la courbe ROC est la courbe paramétrée définie par

$$\begin{aligned} ROC : \mathbb{R} &\to [0,1]^2 \\ s &\mapsto (x(s), y(s)). \end{aligned}$$

Il est possible d'analyser ce graphique en remarquant que l'axe des abscisses donne la spécificité de la règle et l'axe des ordonnées donne la sensibilité de la règle. Cependant, afin d'analyser la courbe ROC, nous définissons deux scores particuliers : le score parfait et le score aléatoire.

Définition 14.2
— *Un score S est dit parfait s'il existe un seuil fini $s^\star$ tel que*

$$\Pr(Y = 1 | S(X) \geq s^\star) = 1 \quad \textit{et} \quad \Pr(Y = 0 | S(X) < s^\star) = 1.$$

— *Un score S est dit aléatoire si les variables $S(X)$ et Y sont indépendantes.*

Ces deux scores sont extrêmes en termes de performance. Si S est parfait, alors la règle de prévision $g_{s^\star}$ ne commettra jamais d'erreur de classification. A l'inverse, si S est aléatoire, alors la note $S(X)$ est indépendante de Y : on ne peut pas faire pire. L'exercice (14.5) indique que la courbe ROC du score parfait correspond donc à la ligne brisée joignant les points $(0,0)$, $(0,1)$, et $(1,1)$. La courbe ROC d'un score aléatoire est le segment qui joint les points $(0,0)$ et $(1,1)$, c'est-à-dire la première bissectrice restreinte au carré $[0,1]^2$ (voir fig. 14.4).

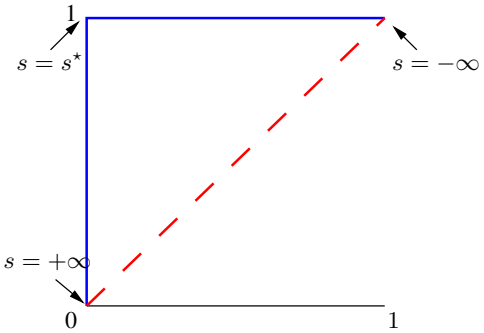


Fig. 14.4 – Courbes ROC d'un score parfait (trait plein) et aléatoire (tirets).

Bien entendu, dans les cas concrets, on est rarement confronté à des scores parfaits et aléatoires. Les performances des scores vont se situer entre ces deux scores extrêmes. Pour le critère ROC, on mesurera donc la performance d'un score par sa proximité avec la courbe ROC du score parfait ou encore par la capacité de sa courbe ROC à se rapprocher le plus rapidement possible de la droite d'équation $y = 1$.

On déduit directement de la courbe ROC un critère numérique qui permet de mesurer la performance d'un score. Il s'agit de l'aire sous la courbe ROC, souvent notée AUC (*Area Under Curve*). Compte tenu des propriétés de la courbe ROC présentées précédemment, il est facile de voir que plus l'AUC d'un score S est important, meilleure est la règle. En particulier, l'AUC vaut 1 pour un score parfait et 0.5 pour un score aléatoire. Ainsi l'AUC d'une règle S va varier entre 0.5 et 1 : plus il sera proche de 1, meilleur sera le score. La propriété suivante donne une interprétation probabiliste de l'AUC.

Proposition 14.1 (Clémençon & Vayatis (2009))
L'AUC d'un score S vérifie

$$AUC(S) = \Pr(S(X) < S(X')|(Y, Y') = (0, 1))$$

où (X', Y') est un couple aléatoire indépendant et de même loi que (X, Y).

L'AUC correspond donc à la probabilité que le score ordonne correctement deux observations prélevées aléatoirement dans les groupes 0 et 1. Cette propriété peut également être utilisée pour calculer la valeur de l'AUC d'un score S (voir exercice 14.7). Il s'agit d'un critère global puisqu'il prend en compte toutes les valeurs de s.

La courbe ROC se calcule à partir des probabilités $x(s)$ et $y(s)$ qui sont en pratique inconnues. Pour un score S donné, ces probabilités pourront être estimées à partir de leurs versions empiriques. En pratique, pour une valeur de s, on calculera la matrice de confusion empirique en précisant la méthode de calcul (voir section 14.2, p. 324) et on en déduira les estimations de $\alpha(s)$ et $1 - \beta(s)$. On répétera ce procédé pour tout un ensemble de s.

14.3.2 Courbe lift

La courbe lift est une autre représentation graphique globale. Cette représentation est dissymétrique et donc le choix du codage (ce que signifie $Y = 1$ et ce que signifie $Y = 0$) a une grande importance ici. La courbe lift est comme la courbe ROC une courbe paramétrée par le seuil s et elle permet de balayer tous les choix de seuil.

Définition 14.3
La courbe lift d'un score $S : \mathbb{R}^p \to \mathbb{R}$ est une courbe paramétrée par le seuil $s \in \mathbb{R}$, de coordonnées suivantes

$$\begin{aligned} x(s) &= \Pr(S(X) > s) \\ y(s) &= \Pr(S(X) > s | Y = 1). \end{aligned}$$

Si nous comparons cette courbe lift à la courbe ROC, à un seuil donné s, elles possèdent la même ordonnée et seule l'abscisse change. L'idée est que la population est ordonnée par le score et l'on regarde quelle est, dans la population des $Y = 1$, la fraction (la probabilité) détectée comme $Y = 1$ en ciblant la partie de la population dont le score est supérieur à s.

Essayons d'interpréter cette courbe lift dans deux exemples. Plaçons-nous dans un exemple de marketing où l'on essaye de prédire si le client est acheteur (noté $Y = 1$) ou non (noté $Y = 0$). Pour cela nous avons des variables censées être prédictives (comme l'âge, le sexe, la CSP, etc.) On construit donc un score S qui permet d'ordonner la population. Ensuite grâce au score estimé sur notre base client, nous enverrons des courriers de sollicitation en envoyant les courriers aux meilleurs scores. On souhaite voir avec la courbe lift l'évolution en fonction de la probabilité/ du pourcentage de clients sollicités dans la population (ou dans la base) quel sera le pourcentage des clients acheteurs qui seront détectés et qui devraient donc acheter suite au mailing.

Plaçons-nous maintenant dans une optique de santé publique : nous avons un test peu coûteux qui permet de diagnostiquer une maladie avec des erreurs. La courbe lift va permettre de voir quelle est l'évolution, en fonction du pourcentage de la population totale soumise au test, du pourcentage chez les malades qui seront effectivement détectés avec le test.

Comme avec la courbe ROC, nous avons deux courbes particulières correspondant à deux règles extrêmes : la courbe du score aléatoire et la courbe du score parfait. La première est sur la première diagonale, alors que pour la seconde, il y a un changement (voir figure 14.5 et l'exercice 14.6, p. 352).

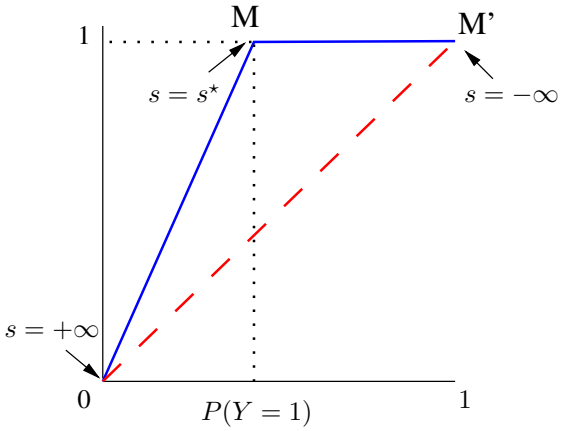


Fig. 14.5 – Courbes lift d'un score parfait (trait plein) et aléatoire (tirets) pour score prenant des valeurs dans $\mathbb{R}$ tout entier.

Bien entendu les courbes lift dans les cas concrets vont se situer entre ces courbes extrêmes.

14.4 Choix du seuil

Le choix du seuil est dépendant du problème dans le sens où tout critère métier pourra prévaloir dans son choix. Il est cependant commun de ne pas en avoir, aussi nous allons lister quelques méthodes classiques de choix de seuil s.

14.4.1 Respect des proportions initiales

La méthode la plus naturelle consiste à choisir un seuil s qui redonne la proportion initiale des 0/1 de l'échantillon. Les probabilités prévues permettent d'ordonner les individus, on choisit alors le seuil qui correspond à la probabilité qui redonne la proportion initiale de 0/1.

14.4.2 Maximisation d'indices ad hoc

Un test (au seuil s) sera d'autant meilleur qu'il présente à la fois une grande sensibilité et une grande spécificité. L'*indice de Younden* J est défini par la somme de ces deux grandeurs moins 1 :

$$J(s) = S_p(s) + S_e(s) - 1.$$

Cet indice varie de 0 (on considère que le test est sans intérêt) à 1 où la règle de décision associée est la règle parfaite. Cet indice donne un poids identique aux Faux Positifs et aux Faux Négatifs.

Il est possible de maximiser le *F1 score* qui est défini comme la moyenne harmonique de la spécificité et de la sensibilité :

$$F_1(s) = \frac{2S_e(s)S_p(s)}{S_e(s) + S_p(s)}.$$

Il est aussi courant de chercher le point de la courbe ROC qui soit le plus près du point $(0, 1)$ (coin gauche). La formule est donnée par

$$\min_s((1 - S_e(s))^2 + (1 - S_p(s))^2).$$

14.4.3 Maximisation d'un coût moyen

Une autre possibilité de choix d'un seuil s est d'introduire une approche minimisant un *coût moyen*. Pour cela, introduisons C_{FP} le coût associé à la détection d'un individu comme positif ($g_s > s$) alors qu'en réalité il est négatif. De même, introduisons C_{FN} le coût associé à la détection d'un individu comme négatif ($g_s \leq s$) alors qu'en réalité il est négatif. Nous avons donc le coût moyen C qui s'écrit :

$$\begin{aligned} C &= C_{\mathrm{FP}} \Pr(g_s > s \cap Y = 0) + C_{\mathrm{FN}} \Pr(g_s \leq s \cap Y = 1) \\ &= C_{\mathrm{FP}} \Pr(g_s > s | Y = 0) \Pr(Y = 0) + C_{\mathrm{FN}} \Pr(g_s \leq s | Y = 1) \Pr(Y = 1) \end{aligned}$$

Dans le milieu médical, il est d'usage d'utiliser la probabilité $\Pr(Y = 1)$ appelée prévalence ; il s'agit simplement de la probabilité d'être malade ($Y = 1$) dans toute la population. Pour des maladies classiques, une bonne estimation de la prévalence est souvent accessible. En introduisant celle-ci, nous obtenons

$$\begin{aligned} C &= C_{\mathrm{FP}}(1 - \Pr(g_s \leq s | Y = 0))(1 - \Pr(Y = 1)) \\ &\qquad + C_{\mathrm{FN}} \Pr(g_s \leq s | Y = 1) \Pr(Y = 1) \\ &= C_{\mathrm{FP}}(1 - S_p(s))(1 - \Pr(Y = 1)) + C_{\mathrm{FN}}(1 - S_e(s)) \Pr(Y = 1). \end{aligned}$$

Ces deux dernières écritures sont complètement équivalentes, la dernière faisant apparaître la spécificité et la sensibilité alors que l'avant-dernière fait apparaître la fonction de répartition conditionnelle chez les $Y = 0$ et $Y = 1$.
Maintenant, choisissons le seuil qui va minimiser le coût moyen et dérivons donc par rapport à s. Rappelons que si la règle de classification admet une densité par rapport à la mesure de Lebesgue (ce que nous supposons ici) la dérivée de la fonction de répartition est la densité et nous avons donc

$$\frac{dC}{ds} = -C_{\mathrm{FP}}(1 - \Pr(Y = 1)) f_{g|Y=0}(s) + C_{\mathrm{FN}} \Pr(Y = 1) f_{g|Y=1}(s).$$

En annulant cette dérivée, nous obtenons l'optimum s^* qui est caractérisé par

$$\frac{f_{g|Y=1}(s^*)}{f_{g|Y=0}(s^*)} = \frac{C_{\mathrm{FP}}}{C_{\mathrm{FN}}} . \frac{1 - \Pr(Y = 1)}{\Pr(Y = 1)}.$$

Ce rapport de densité a une représentation graphique évidente sur la courbe ROC : il s'agit de la pente de la tangente. L'optimum est donc le point (le plus proche du coin gauche) dont la pente vaut la valeur censée être connue : $\frac{C_{\text{FP}}}{C_{\text{FN}}} \cdot \frac{1-\Pr(Y=1)}{\Pr(Y=1)}$.
Si nous prenons les valeurs par défaut en général très peu satisfaisantes de $P(Y = 1) = 0.5$ (la moitié de $Y = 1$ dans la population) et $C_{\text{FP}} = C_{\text{FN}}$ (un coût symétrique quelles que soient les erreurs), alors on cherche le point sur la courbe ROC qui a une tangente de pente 1.
Bien entendu la courbe ROC est estimée et nous cherchons donc en pratique sur la courbe ROC estimée le point dont la pente de la tangente vaut la pente estimée (car en général la prévalence $\Pr(Y = 1)$ est estimée) recherchée.

14.5 Analyse des données chd

14.5.1 Les données

Nous avons déjà commencé à traiter ce jeu de données dans le chapitre sur la régression logistique. Il s'agit d'expliquer la présence ou l'absence de maladie (la variable `chd`). On dispose ici de 9 variables potentiellement explicatives (dont une variable qualitative à 2 modalités) et de 462 individus.

```
> library(bestglm)
Le chargement a nécessité le package : leaps
> data(SAheart)
> maladie <- SAheart
> maladie$chd <- as.factor(maladie$chd)
> dim(maladie)
[1] 462  10
> table(maladie$chd)

  0   1
302 160
```

Il y a donc 160 patients considérés comme malades pour 462 individus de l'étude soit 34 % de malades.
Afin de comparer les différents modèles, nous allons appliquer la validation croisée avec 10 blocs. La présentation que nous allons faire sera un peu lourde (il va y avoir de fortes redondances de codes entre les méthodes) mais devrait permettre de bien comprendre la démarche.

14.5.2 Comparaison des algorithmes

Nous commençons par créer un data-frame qui va contenir les résultats des prévisions obtenues avec chaque méthode. Cet objet va donc admettre 462 lignes qui correspondent aux 462 observations initiales et autant de colonnes que de mé-

thodes. Nous préconisons de rajouter une colonne dédiée à la valeur observée de `Y` ici le `Y`.

```
> SCORE <- data.frame(Y = maladie$Y)
```

Avant d'évaluer par validation croisée 10 blocs, nous effectuons un tirage aléatoire affectant chaque observation $i \in \{1, \ldots, n\}$ dans un bloc $b \in \{1, \ldots, 10\}$. Afin d'obtenir des blocs dont les effectifs sont équilibrés mais dont la proportion de 0/1 est la même que la proportion initiale, nous travaillons en 2 étapes. L'objectif est de créer un vecteur appelé `bloc` qui contiendra dans sa i^{e} coordonnée le numéro du bloc dans lequel le i^{e} individu est affecté.

Créons le vecteur `bloc` rempli de zéro. Puis nous repérons les coordonnées des individus des deux groupes : d'un côté les individus i tels que $Y_i = 0$ et de l'autre ceux qui sont tels que $Y_i = 1$.

```
> nbbloc <- 10
> bloc <- rep(0, nrow(maladie))
> ind0 <- which(maladie$chd==0)
> ind1 <- which(maladie$chd==1)
```

Dans le groupe des $Y_i = 0$, nous effectuons une répartition séquentielle de chaque individu 0 dans les blocs de la manière suivante : la première observation va dans le bloc 1, la seconde dans le bloc 2 ... la 10e dans le bloc 10, la 11e dans le bloc 1, la 12e dans le bloc 2... que nous permutons aléatoirement. Nous faisons ensuite la même chose pour les individus dans le groupe des $Y_i = 1$:

```
> set.seed(1234)
> bloc[ind0] <- sample(rep(1:nbbloc, length = length(ind0)))
> bloc[ind1] <- sample(rep(1:nbbloc, length = length(ind1)))
```

En conclusion, les blocs sont équilibrés en effectif mais aussi en proportion de 0/1. Les méthodes de régression que nous allons comparer sur ces données seront la régression logistique, la régression logistique avec choix de variables, la régression lasso, ridge ou elastic-net.

Régression logistique

Commençons par mettre en œuvre la régression logistique et la régression avec choix de variables en utilisant le package **bestglm**. On conserve dans une liste toutes les variables sélectionnées par bloc pour vérifier si cela reste cohérent.

```
> models <- list()
> for(i in 1:nbbloc){
+  print(i)
+   maladieA <- maladie[bloc!=i, ]
+   maladieT <- maladie[bloc==i, ]
+   ## Logistique global
+   reglog <- glm(chd~., data=maladieA, family="binomial")
+   SCORE[bloc==i,"glm"] <- predict(reglog, maladieT,
+                                                 type="response")
+   ## Logistique choix
+   choix <- bestglm(maladieA,family=binomial)
+   SCORE[bloc==i,"choix"] <- predict(choix$BestModel, maladieT,
+                                                 type="response")
+   tmp <- choix$BestModels[1, -ncol(maladie)]
+   var <- names(tmp)[tmp==TRUE]
+   models[[i]] <- var
+ }
```

Nous avons donc mis en pratique la régression logistique et la régression logistique avec choix de variables. Dans la plupart des blocs, la sélection a conservé les variables `tobacco`, `ldl`, `famhist`, `typea` et `age`. Les résultats des prévisions sont dans le data-frame `SCORE` mais avant d'analyser les résultats, appliquons les régressions logistiques pénalisées.

Lasso, ridge et elastic-net

Dans le chapitre précédent (8.3.1, p. 199), nous avons vu 3 méthodes différentes : ridge, lasso et elastic-net. Pour cette dernière nous allons utiliser comme valeur pour le paramètre α la valeur de 0.5. Il est aussi possible de modifier cette valeur mais nous ne le ferons pas. En préalable, rappelons que **glmnet** utilise des matrices. La fonction **model.matrix** peut être utilisée pour construire la matrice du design `X`. Cette fonction permet d'effectuer le codage des variables explicatives qualitatives sans effort. La constante (fournie en première colonne de la sortie de `model.matrix`) étant traitée directement par `glmnet`, nous la supprimons au passage :

```
> maladie.X <- model.matrix(chd~., data=maladie)[,-1]
> maladie.Y <- maladie[, "chd"]
```

Pour chaque étape de validation croisée, il faut choisir le λ optimal et cela est fait grâce la fonction **cv.glmnet**. Nous effectuons donc une validation croisée dans la validation croisée ! Nous avons vu aussi que le choix de λ peut se faire de différentes manières : déviance (par défaut), erreur de classement (`type.measure="class"`) ou AUC (`type.measure="auc"`). Commençons par la méthode par défaut :

```
> library(glmnet)
> for(i in 1:nbbloc){
+   print(i)
+   XA <- maladie.X[bloc!=i, ]
+   YA <- maladie.Y[bloc!=i]
+   XT <- maladie.X[bloc==i, ]
+   ## ridge
+   mod <- cv.glmnet(XA, YA, alpha=0, family="binomial")
+   SCORE[bloc==i, "ridge"] <- as.vector(
+              predict(mod, XT, "lambda.1se", type="response"))
+   ## lasso
+   mod <- cv.glmnet(XA, YA, alpha=1, family="binomial")
+   SCORE[bloc==i, "lasso"] <-  as.vector(
+              predict(mod, XT, "lambda.1se", type="response"))
+   ## elast
+   mod <- cv.glmnet(XA, YA, alpha=.5, family="binomial")
+   SCORE[bloc==i, "elast"] <-  as.vector(
+              predict(mod, XT, "lambda.1se", type="response"))
+ }
```

Nous avons maintenant un data-frame avec en première colonne la valeur de Y et pour les autres colonnes les probabilités estimées pour les 5 modèles. Voyons les 4 premières lignes de SCORE.

```
> SCORE[1:4, ]
  Y       glm     choix     ridge     lasso     elast
1 1 0.6637242 0.6612646 0.5522532 0.5505895 0.5537659
2 1 0.2974980 0.3695433 0.3308731 0.3786138 0.3645915
3 0 0.2949491 0.3442603 0.3315480 0.3737668 0.3500686
4 1 0.7299084 0.7199016 0.6182603 0.5457484 0.6100731
```

Quels critères choisir pour comparer ces méthodes ? Le choix du critère dépend bien évidemment du problème métier ; il est par exemple possible que dans certains cas on puisse quantifier le coût des faux négatifs FN et des faux positifs FP, mais ce n'est pas le cas ici. Nous allons dans un premier temps utiliser le critère le plus classique : le critère du taux de mal classés.
En appliquant la fonction **tauxMC** (voir l'exercice 14.3 p. 351) avec un seuil classique de $s = 0.5$ on trouve :

```
> round(apply(SCORE[,-1], 2, tauxMC, Y=SCORE$Y, seuil=.5)
                                                / nrow(SCORE) * 100, 2)
  glm choix ridge lasso elast
27.27 27.27 30.30 27.49 27.71
```

Il semblerait que selon ce critère, la régression logistique et la régression logistique

avec choix de variables donnent les meilleurs résultats mais en changeant de seuil (fixé de manière arbitraire à 0.5), il est possible que ce classement évolue.
Nous pouvons également choisir le coefficient λ de la régression logistique pénalisée, non plus par déviance mais par erreur de classement ou par AUC. Cela fait 6 prévisions possibles supplémentaires. Nous ne présentons que le code pour la méthode ridge, mais il est identique pour lasso et elastic-net en remplaçant la valeur de α dans le code.

```
> for(i in 1:nbbloc){
+   print(i)
+   XA <- maladie.X[bloc!=i,]
+   YA <- maladie.chd[bloc!=i]
+   XT <- maladie.X[bloc==i,]
+   ## ridge measure class
+   mod <- cv.glmnet(XA, YA, alpha=0, family="binomial",
+                    type.measure="class")
+   SCORE[bloc==i,"ridgeC"] <- as.vector(
+              predict(mod,XT, "lambda.1se", type="response"))
+   ## ridge measure AUC
+   mod <- cv.glmnet(XA, YA, alpha=0, family="binomial",
+                    type.measure="auc")
+   SCORE[bloc==i,"ridgeA"] <- as.vector(
+              predict(mod,XT, "lambda.1se", type="response"))
+ }
```

Nous avons donc maintenant 11 colonnes de résultats et nous allons comparer les différentes méthodes via leur AUC. Nous ne présentons que les six meilleurs résultats.

```
> library(pROC)
> rocCV <- roc(Y~., data=SCORE)
> sort(round(unlist(lapply(rocCV, auc)), 3), dec=TRUE)[1:6]
 choix    glm elastC  elast ridgeC lassoC
 0.771  0.770  0.769  0.767  0.766  0.765
```

Les résultats sont relativement voisins entre les différentes méthodes. Le choix de la méthode pour déterminer le paramètre λ est peu important sur cet exemple.
Dessinons les 4 courbes ROC correspondant aux 4 meilleures procédures d'estimation en utilisant le code suivant :

```
> ind <- order(aucmodele, decreasing=TRUE)
> mapply(plot, rocCV[ind[1:4]], col=1:4, lty=1:4,
+        lwd=3, legacy.axes=T, add=c(F, T, T, T))
> legend("bottomright", legend=names(SCORE)[ind[1:4]],
+        col=1:4, lty=1:4, lwd=3, cex=1)
```

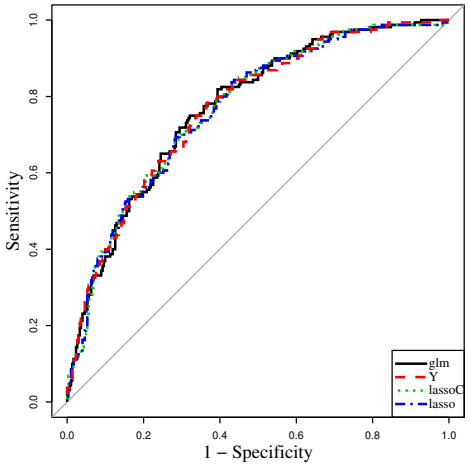


Fig. 14.6 – Les courbes ROC correspondant aux 4 meilleures procédures d'estimation.

Nous retrouvons graphiquement sur la figure 14.6 que les méthodes sont très voisines. Pour chaque méthode, nous résumons les résultats en cherchant le seuil optimal via le critère de Younden ainsi que les valeurs de différents critères pour ce seuil. Nous appliquons donc à chaque composante de la liste `rocCV` la fonction `coords` :

```
> tmp <- lapply(rocCV, FUN =coords, x="best",
+               ret=c("threshold", "tp", "fp", "fn", "tn",
+          "sensitivity", "specificity", "ac"), transpose = TRUE)
> mat <- do.call(rbind, tmp)
```

Les composantes de la liste sont ensuite rassemblées dans la matrice `mat` qui est arrondie :

```
> round(mat,3)
       threshold  tp  fp fn  tn sensitivity specificity accuracy
glm        0.297 127 111 33 191       0.794       0.632    0.688
choix      0.334 120  97 40 205       0.750       0.679    0.703
ridge      0.346 123 101 37 201       0.769       0.666    0.701
lasso      0.311 133 130 27 172       0.831       0.570    0.660
elast      0.377 111  86 49 216       0.694       0.715    0.708
ridgeC     0.349 121  99 39 203       0.756       0.672    0.701
lassoC     0.378 112  90 48 212       0.700       0.702    0.701
elastC     0.380 110  85 50 217       0.688       0.719    0.708
ridgeA     0.346 129 118 31 184       0.806       0.609    0.677
lassoA     0.377 108  89 52 213       0.675       0.705    0.695
elastA     0.371 107  89 53 213       0.669       0.705    0.693
```

Les résultats obtenus sont très proches d'une ligne à l'autre. Notons tout de même que dans tous les cas, la valeur du seuil est inférieure au seuil classique de 0.5. Les critères calculés ici dépendent bien entendu du découpage en 10 blocs effectué. Afin d'obtenir des résultats plus stables, il est courant de répéter cette procédure de validation croisée sur plusieurs découpages en blocs et de faire la moyenne des résultats obtenus. Dans la section suivante, nous proposons donc de répéter la validation croisée plusieurs fois et ensuite de considérer des transformations des variables explicatives dans les modèles.

14.5.3 Pour aller plus loin

Stabilité des résultats

Afin d'obtenir des résultats qui ne dépendent pas du découpage, nous proposons de répéter la procédure de validation croisée 20 fois. Nous allons collecter les résultats dans un **array** à 3 dimensions : la première correspondra aux nombres d'individus, la seconde au nombre de méthodes et la troisième au nombre de validations croisées (ici 20). Nous limitons cet exemple aux 5 premières méthodes vues (logistique, choix de variables, ridge, lasso et elastic-net avec le choix de λ par la méthode de la déviance). On commence par créer tous les objets nécessaires :

```
> library(glmnet) ; library(bestglm)
> data(SAheart)
> maladie <- SAheart
> maladie$chd <- as.factor(maladie$chd)
> maladie.X <- model.matrix(chd~.,data=maladie)[,-1]
> maladie.Y <- maladie[,"chd"]
> ind0 <- which(maladie$chd==0)
> ind1 <- which(maladie$chd==1)
> niter <- 20
```

Ensuite nous créons l'**array** des résultats et

```
> SCORE <- array(0,dim=c(nrow(maladie),5,niter),
+              dimnames=list(1:nrow(maladie),
+                        c("glm","choix","ridge","lasso","elast"),
+                        1:niter))
> nbbloc <- 10
> bloc <- rep(0, nrow(maladie))
```

Il est maintenant possible de remplir l'**array** `SCORE` en faisant 2 boucles : une qui va indiquer quelle validation croisée on effectue (bien penser à faire évoluer la graine de construction des blocs sinon on obtiendra `niter` fois le même résultat) et une qui va gérer les blocs (comme précédemment).

```
for(j in 1:niter){
 set.seed(1233+j)
 bloc[ind0] <- sample(rep(1:nbbloc, length = length(ind0)))
 bloc[ind1] <- sample(rep(1:nbbloc, length = length(ind1)))
 for(i in 1:nbbloc){
   malA <- maladie[bloc!=i,]
   malT <- maladie[bloc==i,]
   ### Logistique global
   reglog <- glm(chd~.,data=malA,family="binomial")
   SCORE[bloc==i,"glm",j] <- predict(reglog,malT,type="response")
   ## Logistique choix
   choix <- bestglm(malA,family=binomial)
   SCORE[bloc==i,"choix",j] <- predict(
                     choix$BestModel, malT, type="response")
   ## matrices apprentissage/test
   XA <- maladie.X[bloc!=i,]
   YA <- maladie.Y[bloc!=i]
   XT <- maladie.X[bloc==i,]
   ## ridge
   mod <- cv.glmnet(XA,YA,alpha=0,family="binomial")
   SCORE[bloc==i,"ridge",j] <- as.vector(
                 predict(mod, XT, "lambda.1se", type="response"))
   mod <- cv.glmnet(XA,YA,alpha=0,family="binomial")
   SCORE[bloc==i,"ridge",j] <- predict(mod,XT,type="response")
   ## la suite est identique (lasso elasticnet),
   ## remplacer alpha et le nom dans l'objet SCORE
 }
}
```

Nous avons donc maintenant 20 matrices de résultats. Pour chaque validation croisée (chaque tranche du **array**), nous calculons l'AUC pour les 5 méthodes et faisons cela 20 fois. Il est ensuite possible de calculer la moyenne des AUC par méthode mais aussi de dessiner un boxplot par méthode.

```
library(pROC)
res <- matrix(0,nrow=niter,ncol=5)
colnames(res) <- dimnames(SCORE)[[2]]
for(j in 1:niter){
   tmp <- SCORE[,,j]
   res[j,]  <- apply(tmp, 2, auc, response=maladie$chd)
}
round(apply(res, 2, mean), 3)
  glm choix ridge lasso elast
0.774 0.774 0.772 0.767 0.770
```

Nous retrouvons globalement les résultats précédents. Analysons la variabilité des résultats via un boxplot

```
> boxplot(res,main="Distribution des AUC par méthode")
```

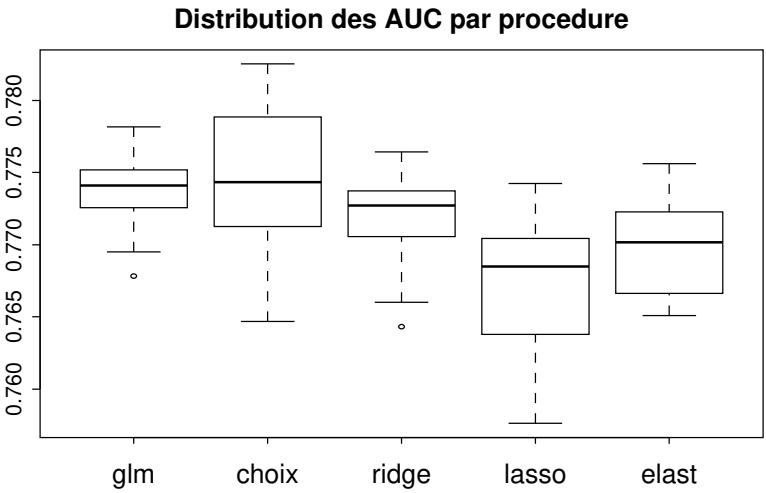


Fig. 14.7 – Boxplot des méthodes.

Les régressions logistiques avec choix de variables donnent les meilleurs résultats mais avec une valeur médiane comparable au modèle de régression logistique classique (avec toutes les variables). Ce dernier est beaucoup plus stable et serait à privilégier. Il faudrait donc estimer le modèle sur toutes les données avant de l'utiliser.

Créations de variables

Nous proposons ici de considérer des transformations des variables explicatives dans les modèles (*features ingeniering*). Commençons par analyser les modèles avec interactions, nous avons maintenant 45 variables, il n'est plus possible d'utiliser **bestglm**, nous utiliserons pour choisir des variables la fonction `step`.

Modèle de prévision avec interactions

Observons qu'il suffit de changer `Y~.` par `Y~.^2` à chaque fois où cela intervient. Nous obtenons alors :

```
SCOREint <- data.frame(Y = maladie$chd)
maladie.X <- model.matrix(chd~.^2,data=maladie)[,-1]
maladie.Y <- maladie[,"chd"]
for(i in 1:nbbloc){
  print(i)
  malA <- maladie[bloc!=i,]
  malT <- maladie[bloc==i,]
  ## Logistique global
  reglog <- glm(chd~.^2,data=malA,family="binomial")
  SCOREint[bloc==i,"glm"] <- predict(reglog,malT,type="response")
  ## Logistique choix
  choix <- step(reglog,trace=0)
  SCOREint[bloc==i,"choix"] <- predict(choix,malT,type="response")
  ## matrices apprentissage/test
  XA <- maladie.X[bloc!=i,]
  YA <- maladie.Y[bloc!=i]
  XT <- maladie.X[bloc==i,]
  ## ridge
  mod <- cv.glmnet(XA,YA,alpha=0,family="binomial")
  SCOREint[bloc==i,"ridge"] <- as.vector(predict(
                      mod, XT, "lambda.1se", type="response"))
  ## la suite est identique (lasso elasticnet),
  ##  remplacer alpha et le nom dans l'objet SCORE
}
> rocCVint <- roc(Y~.,data=SCOREint)
> sort(round(unlist(lapply(rocCVint, auc)),3), decreasing=TRUE)
lasso elast ridge choix   glm
0.776 0.775 0.766 0.743 0.709
```

Les résultats ne sont pas bien meilleurs que ceux obtenus sans les interactions. Poursuivre avec des interactions semble peu utile dans ce problème.

Modèle de prévision avec des polynômes

Nous pouvons facilement construire la matrice (ou un data-frame) avec comme variables, les variables initiales, les variables initiales élevées au carré, les variables initiales élevées au cube. Il faut au préalable déterminer les variables qui sont des facteurs pour ne pas les élever au carré.

```
> typevar <- sapply(maladie, class)
> X <- maladie[, typevar!="factor"]
> maladieP <- data.frame(maladie[,1:9], X^2, X^3, chd=maladie$chd)
> maladie.X <- model.matrix(chd~., data=maladieP)
> maladie.Y <- maladie[, "chd"]
```

Nous avons 25 variables explicatives et les résultats sont ici plus mauvais que les précédents.

```
lasso elast choix   glm ridge
0.768 0.768 0.759 0.756 0.750
```

Cette modification des variables n'a pas eu d'effet et poursuivre la construction de variable à l'aveugle, de manière automatique, ne semble pas opportun.

Modèle de prévision avec des splines

Résumons ce que nous avons appris jusqu'ici : les carrés/cubes des variables semblent inutiles, les interactions aussi. Nous avons vu que certaines variables semblent plus utiles que d'autres (voir section 14.5.2, p. 336). Puisque les variables ne sont pas en grand nombre, nous pouvons examiner les résidus partiels (voir p. 278). Afin de garder toutes les procédures à égalité, nous allons utiliser non pas toutes les données mais seulement les données d'apprentissage, et donc par exemple toutes les données sauf le bloc 1 (on aurait pu tout aussi bien exclure le bloc 2 ou le 3 ; on espère que la structure se conserve d'un bloc à l'autre) :

```
malA <- maladie[bloc!=1, ]
mod <- glm(chd~., data=malA, family="binomial")
```

Comme nous souhaitons trouver d'éventuelles transformations des variables explicatives quantitatives continues, nous éliminons les variables qualitatives et nous calculons les résidus partiels. Ensuite, nous rassemblons pour chaque variable initiale, les valeurs de celle-ci, les résidus partiels et le nom de la variable dans un data-frame :

```
typevar <- sapply(maladie, class)
X <- malA[, typevar!="factor"]
rp <- resid(mod, type="partial")[, colnames(X)]
rpdt <- data.frame(x=as.vector(data.matrix(X)),
                   y=as.vector(rp),
                   z=rep(colnames(rp), each=nrow(rp)))
```

Le couple variable initiale, résidus partiels est ensuite tracé avec un lissage de type spline de lissage (voir section 11.4.3, p. 278 pour la présentation de la méthode).

```
library(lattice)
xyplot(y~x|z,data=rrdt,scale=list(x="free"),
       panel = function(x, y) {
           panel.xyplot(x, y, pch=3)
           smooth <- smooth.spline(x, y)
           panel.lines(x = smooth$x, y = smooth$y) })
```

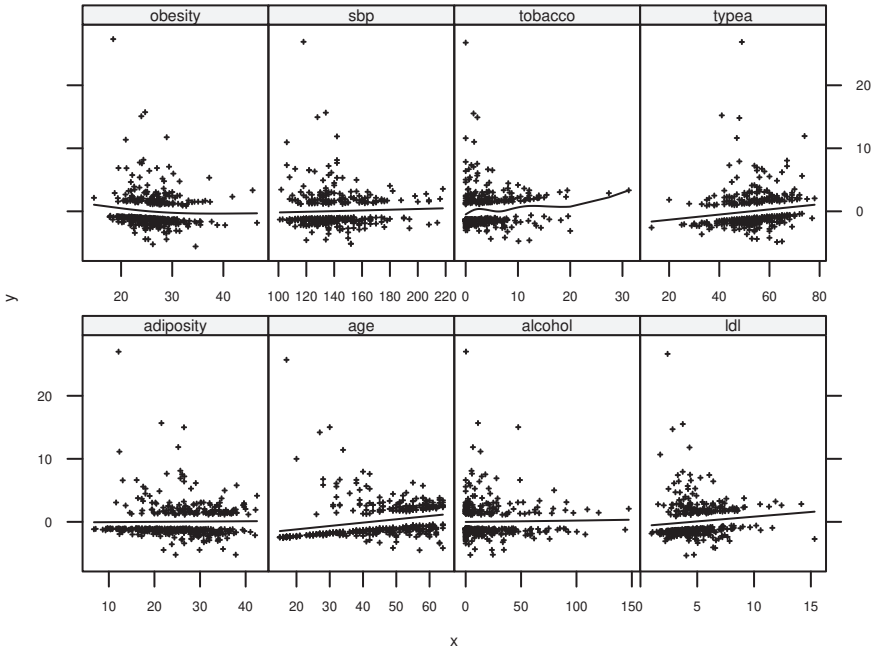


Fig. 14.8 – Résidus partiels.

Nous voyons que la seule variable montrant de la non-linéarité est la variable `tobacco`. Si nous agrandissons le graphique (fig. 14.9), nous voyons que la majeure partie de la non-linéarité se situe dans la première partie de la courbe. Les résidus partiels augmentés donnant les mêmes graphiques, ils ne sont pas représentés ici.

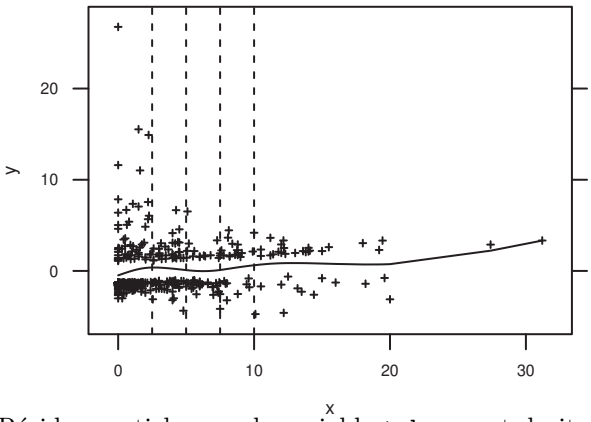


Fig. 14.9 – Résidus partiels pour la variable `tobacco` et droites aux nœuds.

Les splines polynomiales ne sont présentées qu'au chapitre 16 mais il nous semble intéressant de les mettre en application ici. Présentons-les rapidement : il s'agit de bouts de polynômes raccordés en des points que l'on appelle nœuds. Plus on

ajoute de nœuds, plus la spline peut varier (osciller). Nous allons donc placer les nœuds au départ en $2.5, 5, 7.5$ et 10 (ou proposer un autre modèle et enlever le premier nœuds, et garder $5, 7.5$ et 10). Le degré choisi sera le plus petit pouvant permettre des oscillations comme dans la figure 14.9, donc 2.
Générons la base de B-splines :

```
deg <- 2
xi <- c(2.5, 5, 7.5, 10)
K <- length(xi)
var <- maladie$tobacco
BX1 <- bs(var, knots=xi, degree=deg,
               Boundary.knots=c(min(var),max(var)), intercept =F)
colnames(BX1) <- paste("tabac", "b", 1:(K+deg),sep="")
```

Ajoutons cette base aux autres variables explicatives et générons les matrices nécessaires à `glmnet` :

```
maladieS1 <- cbind.data.frame(maladie, BX1)
maladieS1.X <- model.matrix(chd~., data=maladieS1)[, -1]
maladieS1.Y <- maladieS[, "chd"]
```

Nous avons 15 variables explicatives et ce nombre étant assez faible, nous pouvons nous attendre à ce que le lasso, ridge et elastic-net ne soient pas très utiles ici. Par ailleurs, nous pouvons aussi utiliser un modèle plus réduit avec la base de B-splines pour `tobacco` et uniquement les variables `ldl famhist typea` et `age`.

```
glm_bs1_restr glm_bs1 ridge_bs1 choix_bs1 lasso_bs1 elast_bs1
        0.790   0.783     0.782     0.779     0.774     0.771
```

Comme pressenti, nous obtenons les meilleurs résultats avec le modèle comportant le plus petit nombre de variables : la base de B-splines pour `tobacco` et uniquement les variables `ldl famhist typea` et `age`. Si nous supprimons le nœud en 2.5, nous atteignons un AUC de 0.792.

14.6 Application : détection d'images publicitaires sur internet

14.6.1 Les données

On cherche à identifier dans les sites web des images publicitaires. On dispose de 3 279 observations, chaque observation est une image caractérisée par 1 558 attributs (les variables explicatives) et si elle est publicitaire ou pas (variable à expliquer binaire Y : 1 si publicité, 0 sinon). Les données sont disponibles sur le site de l'UCI Machine Learning Repository [2]. On importe les données et on donne

2. https://archive.ics.uci.edu/ml/datasets/internet+advertisements

le nom `Y` à la variable à expliquer :

```
> pub <- read.table("ad_data.txt", header = FALSE, sep = ",",
+                 dec = ".", na.strings = "?", strip.white = TRUE)
> names(pub)[ncol(pub)] <- "Y"
> table(pub$Y)
   ad. nonad.
   459   2820
```

Les effectifs sont très déséquilibrés : dans la base, nous avons seulement 13 % de publicités (`ad.`) à analyser. Nous recodons les modalités `nonad.` en 0 et `ad.` en 1 et nous assignons le niveau de base au niveau 0 avec la fonction `relevel`. Les quatre premières variables potentiellement explicatives admettent un certain nombre de valeurs manquantes (903, 901, 910 et 15). Nous choisissons de supprimer ces variables :

```
> pub$Y <- factor(pub$Y, labels=c("1","0"))
> pub$Y <- relevel(pub$Y, "0")
> pub <- pub[,-(1:4)]
> table(pub$Y)
   0    1
2820  459
```

Nous allons comparer la régression logistique et les régressions pénalisées en utilisant une validation croisée 10 blocs. Nous découpons donc les données en 10 blocs en respectant la proportion de 0/1.

```
> ind0 <- which(pub$Y==0)
> ind1 <- which(pub$Y==1)
```

```
> set.seed(1234)
> nbbloc <- 10 ; bloc <- 1:nrow(pub)
> bloc[ind0] <- sample(rep(1:nbbloc, length = length(ind0)))
> bloc[ind1] <- sample(rep(1:nbbloc, length = length(ind1)))
```

14.6.2 Ajustement des modèles

Au vu du nombre important de variables explicatives, nous nous attendons à ce que les méthodes pénalisées aient de bons résultats. Rappelons que celles-ci choisissent le λ optimal par validation croisée avec la fonction **cv.glmnet**. Nous allons donc effectuer une validation croisée (avec **cv.glmnet**, validation croisée interne) dans notre validation croisée (validation croisée externe). Bien évidemment d'un bloc de validation croisée externe à l'autre, les λ optimaux calculés seront différents et la validation croisée externe permet de choisir une méthode mais pas les paramètres.

Une fois la méthode choisie, il est donc nécessaire de l'appliquer sur toutes les données afin d'avoir un modèle de prévision prêt à l'emploi.

Mettons en place le data-frame qui va contenir les résultats (`SCORE`) ainsi que les matrices utilisées par la fonction `glmnet` :

```
> library(glmnet)
> SCORE <- data.frame(Y=pub$Y)
> pub.X <- model.matrix(Y~., data=pub)[,-1]
> pub.Y <- pub[,"Y"]
```

Nous commençons la boucle sur les blocs et nous réalisons le modèle logistique complet qui sert d'étalon et dont on s'attend à de mauvaises performances.

```
> for(i in 1:nbbloc){
+  print(i)
+  pubA <- pub[bloc!=i,]
+  pubT <- pub[bloc==i,]
+  ## Logistique global
+  reglog <- glm(Y~., data=pubA, family="binomial")
+  SCORE[bloc==i,"glm"] <- predict(reglog, pubT, type="response")
```

Ensuite nous effectuons un choix de variables ascendant ; le nombre de variables explicatives étant très grand, partir d'un petit nombre de variables semble plus judicieux :

```
+  ###Logistique choix
+  reg1 <- glm(Y~1,data=pubA,family="binomial")
+  choix <- step(reg1,scope=list(lower=formula(reg1),
                                 upper=formula(reglog)),trace=0)
+  models[[i]] <- formula(choix)
+  SCORE[bloc==i,"glmch"] <- predict(choix,pubT,type="response")
```

Enfin, pour les méthodes pénalisées, nous avons vu que le choix de λ peut se faire de différentes manières : déviance (par défaut), erreur de classement (avec l'option `type.measure="class"`) ou AUC (avec l'option `type.measure="auc"`). Nous donnons le code pour la logistique, ridge, lasso et elastic-net avec le choix classique de λ. Pour les autres critères de choix de λ, le code n'est pas donné mais il se trouve dans l'exemple précédent.

```
+  ## méthodes pénalisées
+  XA <- pub.X[bloc!=i,]
+  YA <- pub.Y[bloc!=i]
+  XT <- pub.X[bloc==i,]
+  ## ridge
+  mod <- cv.glmnet(XA, YA, alpha=0, family="binomial")
+  SCORE[bloc==i,"ridge"] <- as.vector(predict(
+                        mod, XT, "lambda.1se", type="response"))
+  ##  lasso
+  mod <- cv.glmnet(XA, YA, alpha=1, family="binomial")
+  SCORE[bloc==i,"lasso"] <- as.vector(predict(
+                        mod, XT, "lambda.1se", type="response"))
+  ##  elast
+  mod <- cv.glmnet(XA, YA, alpha=.5, family="binomial")
+  SCORE[bloc==i,"elast"] <- as.vector(predict(
+                        mod, XT, "lambda.1se", type="response"))
}
```

Nous pouvons donc maintenant comparer les 10 différentes modélisations. Cette comparaison va être faite via l'AUC ; nous affichons les 6 meilleures méthodes :

```
> library(pROC)
> rocCV <- roc(Y~.,data=SCORE)
> sort(round(unlist(lapply(rocCV, auc)), 3), decreasing=TRUE)[1:6]
 ridge ridgeC ridgeA elastA lassoA elastC
 0.982  0.982  0.963  0.963  0.961  0.959
```

Quelle que soit la façon de choisir λ, la méthode ridge donne de bien meilleurs résultats que les autres méthodes. Cet exemple permet de mettre en évidence l'apport des méthodes régularisées en grande dimension.

Si l'on répète 20 validations croisées (attention à la durée de l'exécution des codes) en parallélisant.

```
> library(doParallel)
> niter <- 20
> coredispo <- max(detectCores()-1,1)
> cl <- makeCluster(min(coredispo,niter))
> clusterSetRNGStream(cl,iseed=78)
> registerDoParallel(cl)
```

```
> res <- foreach(j=1:niter) %dopar% {
+   SCORE <- data.frame(Y=pub$Y)
+   nbbloc <- 10
+   set.seed(1234+j)
+   bloc[ind0] <- sample(rep(1:nbbloc, length = length(ind0)))
+   bloc[ind1] <- sample(rep(1:nbbloc, length = length(ind1)))
+   for(i in 1:nbbloc) {
     ... }
    SCORE
+ }
```

La liste des 20 data-frames `SCORE` est ensuite analysée en appliquant les fonctions `ffroc` puis `ffauc` à chaque composante. Enfin les résultats (AUC) sont rassemblés dans une matrice (`aucm4_20`) et on en tire la moyenne :

```
> ffroc  <- function(x) roc(Y~., data=x)
> ttm4_20 <- lapply(scorepubm4_20VC, ffroc)
> ffauc  <- function(x) unlist(lapply(x, auc))
> aucm4_20 <- do.call(rbind, lapply(ttm4_20, ffauc))
> print(sort(apply(aucm4_20,2,mean), decreasing=TRUE)[1:6], 3)
 ridge ridgeC elastA ridgeA lassoA  elast
 0.980  0.979  0.963  0.963  0.959  0.954
```

Nous retrouvons globalement les résultats précédents. Analysons la variabilité des résultats via un boxplot.

```
> boxplot(res)
```

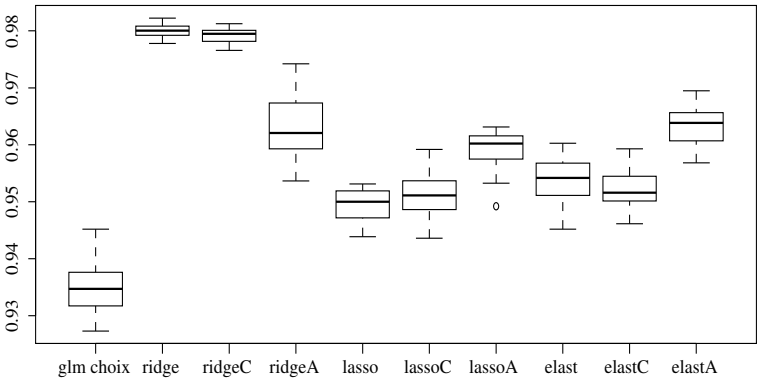


Fig. 14.10 – Boxplot des méthodes sans le modèle complet logistique.

Les régressions logistique ridge restent les meilleures et leurs AUC très proches de 1 nous indiquent que cette modélisation est difficilement améliorable. Nous nous arrêterons donc ici.

14.7 Exercices

Exercice 14.1 (Questions de cours)

1) La spécificité mesure
   A. la probabilité de classer à tort un label $Y = 0$,
   B. la probabilité de classer à tort un label $Y = 1$,
   C. la probabilité de classer à raison un label $Y = 0$,
   D. la probabilité de classer à raison un label $Y = 1$.
2) La sensibilité mesure
   A. la probabilité de classer à tort un label $Y = 0$,
   B. la probabilité de classer à tort un label $Y = 1$,
   C. la probabilité de classer à raison un label $Y = 0$,
   D. la probabilité de classer à raison un label $Y = 1$.
3) L'acronyme AUC désigne
   A. l'aire sous la courbe ROC,
   B. la courbe paramétrée $s \mapsto (\alpha(s), \beta(s))$,
   C. la courbe paramétrée $s \mapsto (\alpha(s), 1 - \beta(s))$.
4) L'acronyme ROC désigne
   A. l'aire sous la courbe AUC,
   B. la courbe paramétrée $s \mapsto (\alpha(s), 1 - \beta(s))$,
   C. la courbe paramétrée $s \mapsto (\alpha(s), \beta(s))$.

Exercice 14.2 († Règle de Bayes)

1) Soit (X, Y) un couple aléatoire à valeurs dans $\mathbb{R}^p \times \{0, 1\}$.
   a) Rappeler la définition de la règle de Bayes $g^\star$ et de l'erreur de Bayes $L^\star$.
   b) Soit g une règle de décision. Montrer que

$$\Pr(g(X) \neq Y | X = x) = 1 - (\mathbf{1}_{g(x)=1}\eta(x) + \mathbf{1}_{g(x)=0}(1 - \eta(x)))$$

   où $\eta(x) = \Pr(Y = 1 | X = x)$.
   c) En déduire que pour tout $x \in \mathbb{R}^p$ et pour toute règle g

$$\Pr(g(X) \neq Y | X = x) - \Pr(g^\star(X) \neq Y | X = x) \geq 0.$$

   d) Conclure.
2) On considère (X, Y) un couple aléatoire à valeurs dans $\mathbb{R} \times \{0, 1\}$ tel que

$$X \sim \mathcal{U}[-2, 2], \qquad U \sim \mathcal{U}[0, 10] \quad \text{et} \quad Y | X = x = \begin{cases} \mathbf{1}_{U \leq 2} & \text{si } x \leq 0 \\ \mathbf{1}_{U > 1} & \text{si } x > 0 \end{cases}$$

   où $\mathcal{U}[a, b]$ désigne la loi uniforme sur $[a, b]$. Les variables X et U sont supposées indépendantes. Calculer la règle de Bayes et l'erreur de Bayes.

Exercice 14.3 (Fonctions R)

En vous inspirant du code précédent et des codes déjà proposés, écrivez

1. une fonction **tauxMC** qui prend en argument les valeurs de Y, une probabilité estimée p et un seuil s et calcule le taux de mal classés. Appliquer ensuite cette fonction à l'objet `SCORE` en jouant avec la valeur de s ;

2. une fonction **sensibilite** qui prendrait les mêmes arguments que la **tauxMC**. Visualiser les résultats de cette fonction pour les différentes méthodes en faisant varier la valeur de s.

Exercice 14.4 (Score aléatoire)
Considérons une règle S aléatoire, montrer que la courbe ROC de cette règle est le segment qui joint les points $(0,0)$ et $(1,1)$, c'est-à-dire la première bissectrice restreinte au carré $[0,1]^2$.

Exercice 14.5 (Score parfait et courbe ROC)
Considérons une règle S parfaite et son seuil associé $s^\star$, montrer que la courbe ROC passe

1. par le point $(0,1)$ quand $s = s^\star$,
2. par les points $[(0,1),(0,0)]$ quand $s > s^\star$,
3. par les points $[(1,1),(0,1)]$ quand $s < s^\star$.

La courbe ROC de la règle parfaite correspond à la ligne brisée joignant les points $(0,0)$, $(0,1)$, et $(1,1)$.

Exercice 14.6 (Score parfait et courbe lift)
Considérons une règle S parfaite et son seuil associé $s^\star$.

1. Montrer que la courbe lift passe par le point $(\Pr(Y=1),1)$ quand $s = s^\star$.

2. Montrer que la courbe est le segment $[(\Pr(Y=1),1),(0,0)]$ quand $s > s^\star$. Pour cela, retrouver qu'un point quand $s > s^\star$ à pour coordonnées $x(s), y(s)$ définies par $x(s) = \Pr(S(X) > s)$ et $y(s) = x(s)/\Pr(Y=1)$. En déduire qu'il s'agit bien d'un segment dont on donnera les extrémités et la pente.

3. Montrer que la courbe lift passe par les points $[(1,1),(\Pr(Y=1),1)]$ quand $s < s^\star$. Pour cela, remarquer que $\forall s < s^\star$, tous les $Y = 1$ sont détectés. Trouver ensuite le point atteint pour le seuil minimal (0 dans le cas d'un score issu d'une probabilité).

La courbe lift de la règle parfaite correspond donc à la ligne brisée joignant les points $(0,0)$, $(\Pr(Y=1),1)$, et $(1,1)$.

Exercice 14.7 (Calcul de scores avec R)
On considère n observations $(X_1,Y_1),\ldots,(X_n,Y_n)$ où X_i est à valeurs dans $\mathbb{R}^p$ et Y_i dans $\{0,1\}$. On se donne également $S : \mathbb{R}^p \to \mathbb{R}$ une fonction de score.

1) A partir de la proposition 14.1, proposer un estimateur de l'AUC de S.
2) On considère les données de l'exercice 11.6.
   a) Séparer l'échantillon en un échantillon d'apprentissage de taille 300 et un échantillon de validation de taille 200.
   b) Estimer les paramètres du modèle logistique avec les données d'apprentissage uniquement. En déduire une fonction de score $\hat{S}$.
   c) Calculer les valeurs de score pour chaque individu de l'échantillon de validation.
   d) A partir de l'estimateur proposé à la question 1), calculer $\mathrm{AUC}(\hat{S})$.
   e) Retrouver $\mathrm{AUC}(\hat{S})$ à l'aide de la fonction **roc** du package **pROC**.

Exercice 14.8 († Validation croisée)
On considère le jeu de données sur les images publicitaires présenté dans la section 14.6. On souhaite ici comparer la régression logistique, les régressions ridge, lasso et elastic-net par validation croisée.

1) Séparer l'échantillon en 10 blocs possédant approximativement la même taille. On pourra utiliser la fonction **sample**.
2) A l'aide de l'algorithme de validation croisée, calculer pour chaque méthode la probabilité estimée qu'une image soit une publicité.
3) En déduire les courbes ROC, les AUC et les erreurs de classification des 4 méthodes.

Chapitre 15

Données déséquilibrées

Ce chapitre traite d'un problème très fréquemment rencontré en classification binaire : les *données déséquilibrées*. Il se produit lorsque les deux modalités de la variable cible Y ne sont pas représentées de façon égale ou plus précisément lorsqu'une des deux modalités est fortement majoritaire. On peut citer comme exemples les cas de la détection de fraudes (peu de fraudeurs), de la présence d'une maladie rare (peu de patients atteints)... Les algorithmes standards peuvent être mis en difficulté et de nouvelles stratégies doivent être élaborées. Nous proposons d'étudier tout d'abord le problème du déséquilibre à travers l'exemple du modèle logistique. Cette étude nous permettra de mieux identifier les difficultés. Nous mettrons ensuite en place différentes stratégies pour répondre à ce problème. Elles consisteront à utiliser des critères de performance qui permettent de prendre en compte le déséquilibre et/ou à ré-échantillonner les données pour se rapprocher d'une situation équilibrée. Nous discuterons enfin du choix de la meilleure stratégie pour un problème de données déséquilibrées.

15.1 Données déséquilibrées et modèle logistique

15.1.1 Un exemple

Nous considérons deux modèles logistiques univariés où la variable explicative X suit une loi uniforme sur $[0, 1]$. Les caractéristiques de ces modèles sont :

$$\begin{aligned} \mathsf{logit}\, p_{\beta_{mod1}}(x_i) &= 3 - 6x_i, \quad \text{modèle 1} \\ \mathsf{logit}\, p_{\beta_{mod2}}(x_i) &= 1 - 6x_i, \quad \text{modèle 2.} \end{aligned}$$

Nous proposons d'étudier numériquement les propriétés de l'estimateur du maximum de vraisemblance (EMV) de $\beta = (\beta_1, \beta_2)'$ dans ces deux modèles. Naïvement, les deux modèles ne différant que par la valeur de β_1, on s'attend à ce que les estimations de β_2 se comportent de manière similaire. Avant de simuler des Y suivant une loi de Bernoulli de paramètre $p_\beta(x_i)$ pour ces deux modèles, visualisons l'évolution de $p_\beta(x)$ en fonction de x.

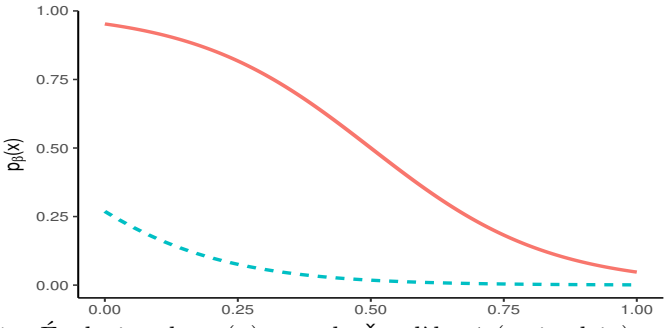


Fig. 15.1 – Évolution de $p_\beta(x)$ pour les modèles 1 (trait plein) et 2 (tirets).

On remarque que $p_\beta(x)$ varie entre 0 et 1 avec une symétrie par rapport à 0.5 pour le modèle 1 alors que, pour le second modèle, $p_\beta(x)$ prend toujours des valeurs faibles : la plus forte probabilité vaut 0.25 quand x_i est proche de zéro. D'un point de vue pratique, cela signifie que dans les échantillons issus du modèle 1, les proportions de 0 et de 1 seront proches alors que dans le cas du modèle 2 nous devrions avoir beaucoup plus de 0.
Nous générons 1000 échantillons de 200 points ($i = 1, \ldots, n$) selon ces deux modèles. Comme annoncé, dans le modèle 1 les proportions de 0 et de 1 sont (quasiment) équilibrées alors que dans le modèle 2, nous observons que la proportion de 1 est de 5 % en moyenne. Nous calculons pour chaque modèle les estimateurs du maximum de vraisemblance de β_1 et β_2. Nous obtenons ainsi, pour chaque modèle, mille valeurs pour $\hat{\beta}_1$ et $\hat{\beta}_2$. On visualise sur la figure 15.2 la distribution de ces valeurs à l'aide d'un boxplot, la vraie valeur de β a été retranchée afin de pouvoir mieux visualiser le biais.

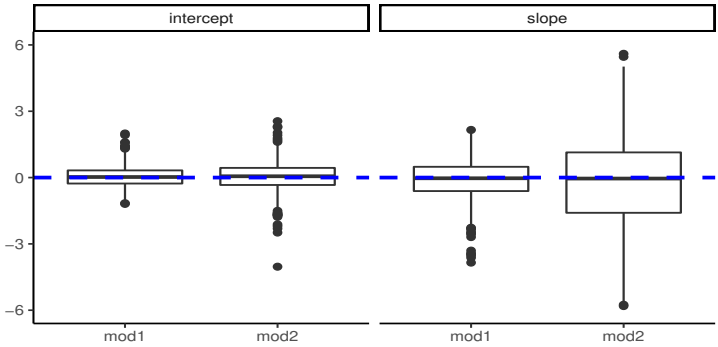


Fig. 15.2 – Boxplot de $\hat{\beta}_1 - \beta_1$ (gauche) et $\hat{\beta}_2 - \beta_2$ (droite) pour les 2 modèles.

Nous observons des performances différentes pour les estimateurs. Le biais est faible dans tous les cas mais la variance est beaucoup plus élevée dans le second modèle, surtout pour l'estimation de β_2. Ce constat peut paraître surprenant

puisque la seule différence entre ces deux modèles est la valeur de β_1. Le déséquilibre contribue à cette augmentation de variance.
Revenons aux propriétés de l'EMV présentées dans la section 11.2.3. Nous avons vu que, pour n suffisamment grand, la matrice de variance de l'EMV peut être approchée par

$$\mathcal{I}_n(\hat{\beta})^{-1} = (X'W_{\hat{\beta}}X)^{-1}$$

où $W_{\hat{\beta}}$ est la matrice diagonale $n \times n$ de terme général $p_{\hat{\beta}}(x_i)(1 - p_{\hat{\beta}}(x_i)), i = 1, \ldots, n$. Lorsque nous sommes en présence de données déséquilibrées, les probabilités $p_{\hat{\beta}}(x_i)$ ont tendance à approcher 0 ou 1 et par conséquent les termes de la diagonale de $W_{\hat{\beta}}$ se rapprochent de 0. Un terme nul sur la diagonale de $W_{\hat{\beta}}$ équivaut à supprimer l'observation associée à ce terme. Nous avons ici beaucoup de valeurs à quasiment 0 et donc nous supprimons « quasiment » un grand nombre d'observations et la variance augmente.
Le déséquilibre influence donc la variance de l'EMV et les estimateurs sont moins précis. Une solution classique pour pallier cette difficulté consiste à rééquilibrer les données, c'est-à-dire à « s'arranger » pour avoir des proportions de 0 et de 1 du même ordre dans l'échantillon. Bien entendu, un tel rééquilibrage impacte le modèle : on n'estime plus les mêmes quantités lorsqu'on calcule les estimateurs sur des données rééquilibrées ! Il est donc nécessaire de bien expliciter la façon dont le rééquilibrage est effectué afin de s'assurer que les conclusions soient correctes. Dans le cadre du modèle logistique, il est possible de prendre en compte le rééquilibrage dans l'écriture du modèle. Nous l'expliquons dans la partie suivante.

15.1.2 Rééquilibrage pour le modèle logistique

Pour simplifier les notations, nous considérons le cas d'une seule variable explicative mais rien ne change dans le cas où il y en a plusieurs. Pour un couple (x, y) où $x \in \mathbb{R}$ désigne la variable explicative et y la variable à expliquer à valeurs dans $\{0, 1\}$, on considère le modèle logistique

$$\mathsf{logit}\, p_\beta(x) = \beta_1 + \beta_2 x. \tag{15.1}$$

Le problème est toujours d'estimer $\beta = (\beta_1, \beta_2)'$. On se place dans le cas de données déséquilibrées et, pour fixer les idées, on considère la situation où $p_\beta(x)$ est petit. Dans ce cas la proportion de 1 dans l'échantillon est faible et la variance de l'EMV de β_2 risque d'être élevée. L'idée va donc être d'essayer d'augmenter cette proportion de 1 afin d'améliorer la performance de l'EMV. Nous introduisons alors une nouvelle variable s à valeurs dans $\{0, 1\}$ qui va permettre de donner plus de poids à la classe minoritaire dans l'étude. Le contrôle de ce poids s'effectue en définissant la loi de s conditionnellement à y à travers les probabilités conditionnelles

$$\tau_0 = \Pr(s = 1|y = 0, x) \quad \text{et} \quad \tau_1 = \Pr(s = 1|y = 1, x).$$

Il existe plusieurs façons de construire cette variable qui seront présentées dans la section 15.1.3. Mais à ce stade on pourra garder en tête que si l'objectif est d'augmenter le poids de la classe 1, on choisira τ_1 grand et τ_0 petit. Remarquons que τ_0

et τ_1 ne dépendent pas de x, cela revient à dire que s et x sont, conditionnellement à y, indépendants.
Un individu est donc désormais constitué d'un triplet (x, y, s) et le modèle rééquilibré va se concentrer uniquement sur les individus pour lesquels s vaut 1. Cela signifie que ce modèle va s'intéresser à la loi de $y|s = 1$. Cette loi est toujours une loi de Bernoulli mais rien n'indique que son paramètre (son espérance) soit toujours $p_\beta(x)$. On désigne donc par $p_\gamma(x) = \Pr_\gamma(y = 1|s = 1, x)$ le paramètre de cette loi et on s'intéresse au lien entre $p_\gamma(x)$ et $p_\beta(x)$. Ce lien est présenté dans la propriété suivante dont on pourra faire la preuve dans l'exercice 15.2.

Proposition 15.1
On a

$$\mathsf{logit}\, p_\gamma(x) = \mathsf{logit}\, p_\beta(x) + \log\left(\frac{\tau_1}{\tau_0}\right). \tag{15.2}$$

Cette propriété est très intéressante car elle implique que, à une constante additive près, les logit des probabilités $p_\beta(x)$ et $p_\gamma(x)$ sont identiques. Cela signifie qu'il sera facile de récupérer les estimateurs de β à partir d'estimateurs calculés sur un échantillon rééquilibré. En effet, si on considère le modèle logistique pour les données rééquilibrées

$$\mathsf{logit}\, p_\gamma(x) = \gamma_1 + \gamma_2 x$$

alors on déduit de la propriété précédente

$$\beta_1 = \gamma_1 - \log\left(\frac{\tau_1}{\tau_0}\right) \quad \text{et} \quad \beta_2 = \gamma_2.$$

Ainsi seule la constante est affectée par le rééquilibrage. De plus, la formule précédente nous apprend comment corriger la constante. En effet, une fois l'EMV $\hat{\gamma}$ obtenu sur les données rééquilibrées, on peut estimer β_2 par $\hat{\gamma}_2$ et β_1 par

$$\hat{\gamma}_{1,\text{cor}} = \hat{\gamma}_1 - \log\left(\frac{\tau_1}{\tau_0}\right). \tag{15.3}$$

Remarque
L'étude a été présentée pour une seule variable explicative. Rien ne change si on en a plusieurs, seule la constante est affectée par le rééquilibrage et elle se corrige de la même façon. Par ailleurs, cette méthode de rééquilibrage peut s'appliquer avec d'autres estimateurs que l'EMV, par exemple avec les estimateurs ridge, lasso... Le protocole est le même : le biais produit par le rééquilibrage est concentré sur la constante qui se corrige de la même façon.

15.1.3 Exemples de schéma de rééquilibrage

Nous présentons des cas d'application des résultats de la section précédente. Commençons par montrer empiriquement l'intérêt du rééquilibrage sur un exemple simulé, mais irréaliste en pratique. Nous présentons ensuite des situations plus concrètes avec les études cas/témoins et les méthodes de sous/sur-échantillonnage.

Cas simulé : retour à l'exemple initial

On propose un schéma de rééquilibrage pour estimer les paramètres du modèle 2 de l'introduction (le modèle déséquilibré)

$$\mathsf{logit}\, p_\beta(x) = \beta_1 + \beta_2 x$$

où x est uniforme sur $[0, 1]$, $\beta_1 = -1$ et $\beta_2 = -6$. Afin de comparer les performances des EMV calculés sur des données brutes (déséquilibrées) et rééquilibrées, on va considérer des échantillons aux caractéristiques identiques : même nombre d'observations $n = 200$ et mêmes points de design $x_1, \ldots, x_n$ de loi uniforme sur $[0, 1]$. On calcule ainsi les EMV sur des échantillons

1. « classique » : $(x_1, y_1), \ldots, (x_n, y_n)$ tels que les y_i sont générés selon des lois de Bernoulli $p_\beta(x_i)$. On note $\hat{\beta}_1$ et $\hat{\beta}_2$ les EMV obtenus.
2. « rééquilibré » : $(x_1, y_1, 1), \ldots, (x_n, y_n, 1)$ où les y_i sont obtenus en faisant :

a) s_i <- 0
b) Tant que $s_i = 0$
   i. générer y_i selon le modèle (15.1)
   ii. générer s_i selon une loi de Bernoulli de paramètre τ_{y_i}
c) Renvoyer y_i.

On utilise ici

$$\tau_0 = \Pr(s_i = 1 | y_i = 0, x_i) = 0.05 \quad \text{et} \quad \tau_1 = \Pr(s_i = 1 | y_i = 1, x_i) = 0.95,$$

et on note $\hat{\gamma}_1$ et $\hat{\gamma}_2$ les EMV calculés.

On répète l'expérience 1 000 fois et on visualise les distributions des estimateurs sur la figure 15.3.

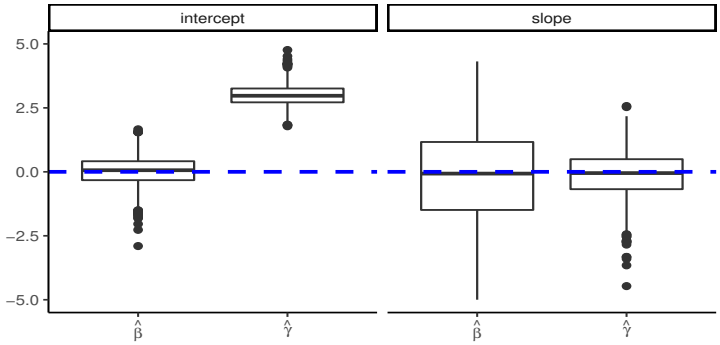


Fig. 15.3 – Distribution de $\hat{\beta}_j - \beta_j$ et $\hat{\gamma}_j - \beta_j$ pour $j = 1$ (gauche) et 2 (droite).

Les proportions de 1 sont aux alentours de 5 % dans les échantillons initiaux contre 30 % dans les échantillons rééquilibrés. On remarque que l'estimateur de

β_2 est nettement plus précis avec les échantillons rééquilibrés. Comme attendu, il possède beaucoup moins de variance. En revanche, on observe un biais élevé pour les estimateurs de β_1 calculés sur les échantillons rééquilibrés. C'est normal puisqu'on n'a pas appliqué ici la correction (15.3). Le graphique 15.4 permet de visualiser la précision de l'estimateur corrigé.

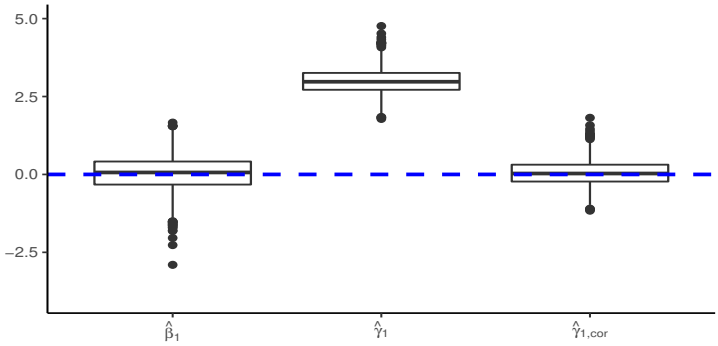


Fig. 15.4 – Distributions de $\hat{\beta}_1 - \beta_1$, $\hat{\gamma}_1 - \beta_1$ et $\hat{\gamma}_{1,\text{cor}} - \beta_1$.

Cas pratique : études cas/témoins et sous/sur-échantillonnages

Bien entendu, la manière de générer les échantillons rééquilibrés que nous avons utilisée dans la simulation précédente ne peut être que rarement appliquée en pratique. Il n'est en effet presque jamais possible de générer plusieurs valeurs de Y en chaque point de design. On peut néanmoins obtenir des données rééquilibrées et appliquer cette technique de correction de la constante dans certaines situations. Un exemple classique est celui des études *cas/témoins* très souvent utilisées en épidémiologie. En effet dans ce type d'étude l'échantillonnage s'effectue en choisissant des individus dans la population des patients sains ($y = 0$) et atteints ($y = 1$) et en étudiant, pour les patients sélectionnés, différentes caractéristiques pouvant potentiellement expliquer la pathologie. Considérons par exemple le cas où, pour une maladie rare, on souhaite modéliser l'influence du tabac sur la maladie. On sait que, pour cette pathologie, la probabilité d'être atteint dans la population générale est de 5 sur 1 000. On réalise une étude cas/témoins en choisissant 250 cas (malade) et 250 témoins (non malade). Parmi les patients sélectionnés, on regarde si la patient est fumeur. Les données sont présentées dans le tableau suivant :

	Malade	Non malade
Fumeur	208	48
Non fumeur	42	202

Tableau 15.1 – Résultats de l'étude.

On considère un modèle logistique permettant d'expliquer la pathologie par le fait

de fumer :

$$\mathsf{logit}(p_\gamma(x)) = \gamma_1 + \gamma_2 \mathbb{1}_{x=\text{FUMEUR}}$$

où $p_\gamma(x)$ désigne la probabilité d'être `MALADE` pour l'individu x. On peut estimer ce modèle avec les commandes suivantes

```
> df <- data.frame(MALADE=c(208,42),NON_MALADE=c(48,202),
                                FUMEUR=c("OUI","NON"))
> model <- glm(cbind(MALADE, NON_MALADE)~FUMEUR,data=df,
                                family=binomial)
> coef(model)
  (Intercept)   FUMEUROUI
    -1.570598    3.036935
```

et en déduire les probabilités d'être atteint par la pathologie avec

```
> newX <- data.frame(FUMEUR=c("OUI","NON"))
> rownames(newX) <- c("OUI","NON")
> predict(model,newdata=newX,type="response")
      OUI       NON
0.8125000 0.1721311
```

Il faut être vigilant avec l'interprétation de ces probabilités. Elles estiment les probabilités d'être malade dans une population où on a autant de malades (cas) que d'individus sains (témoins). Ce n'est pas le cas dans la population générale puisque l'on sait que 5 individus sur 1 000 sont atteints. Si on souhaite une estimation pour la population générale, il faut corriger la constante en utilisant la propriété 15.1. Il suffit pour cela de calculer le rapport τ_1/τ_0. Si on désigne par s la variable qui vaut 1 si un individu est dans l'échantillon et 0 sinon, on a

$$\tau_j = \Pr(s=1|y=j) = \frac{\Pr(y=j|s=1)\Pr(s=1)}{\Pr(y=j)}, \quad j=0,1.$$

Comme nous avons choisi un échantillon équilibré, nous avons $\Pr(y=1|s=1) = \Pr(y=0|s=1)$. Nous déduisons

$$\frac{\tau_1}{\tau_0} = \frac{\Pr(y=0)}{\Pr(y=1)} = \frac{0.995}{0.005}.$$

Ainsi, si on désigne par $p_\beta(x)$ la probabilité, pour un individu x de la population générale, d'être `MALADE`, on a

$$\mathsf{logit}(p_\beta(x)) = \gamma_1 - \log\left(\frac{0.995}{0.005}\right) + \gamma_2 \mathbf{1}_{x=\text{FUMEUR}}.$$

Il reste donc maintenant à corriger la constante manuellement avec

```
> beta1_cor <- coef(model)[1]-log(0.995/0.005)
> beta2 <- coef(model)[2]
```

On peut ensuite en déduire, en utilisant le modèle, les probabilités d'être atteint par la pathologie en étant fumeur et non-fumeur respectivement :

```
> exp(beta1_cor+beta2)/(1+exp(beta1_cor+beta2))
 0.02131148
> exp(beta1_cor)/(1+exp(beta1_cor))
0.001043738
```

Un autre cas classique de rééquilibrage est celui du sous-échantillonnage ou du sur-échantillonnage. Considérons par exemple le cas où on dispose d'un échantillon de grande taille déséquilibré. Un rééquilibrage naturel revient à tirer avec ou sans remise avec les taux de sondage τ_0 et τ_1 des individus parmi les positifs et négatifs. Cela peut se faire aisément en R sur un tableau `df` dans lequel la variable à expliquer est `Y` à l'aide de la fonction **sample** :

```
> tau <- c(0.05,0.95)
> ind0 <- which(df$Y==0)
> ind1 <- which(df$Y==1)
> choix0 <- sample(ind0, size=length(ind0)*tau[1], replace = F)
> choix1 <- sample(ind1, size=length(ind1)*tau[2], replace = F)
> dff <- rbind(df[choix0,], df[choix1,])
```

On peut alors ajuster le modèle logistique sur l'échantillon rééquilibré

```
> mod <- glm(Y~., data=dff, family="binomial")
```

et corriger la constante avec

```
> gamma <- coef(mod)
> gamma[1] - log(tau[2]/tau[1])
```

La correction de la constante peut se faire en utilisant la fonction **offset**

```
> glm(Y~. + offset(rep(log(tau[2]/tau[1]), nrow(dff))),
+     data=dff, family="binomial")
```

On peut remarquer que, sous réserve d'avoir bien choisi τ_0 et τ_1, *l'échantillon* `dff` *est effectivement équilibré mais possède moins d'observations que l'échantillon initial.* Par conséquent, rien ne garantit que le modèle ajusté sur cet échantillon soit plus performant que le même modèle ajusté sur les données initiales. C'est pourquoi il est recommandé de toujours comparer les performances de ces modèles comme nous le ferons dans la section 15.3. Enfin la technique utilisée ici pour rééquilibrer a consisté à sous-échantillonner en tirant des observations au hasard parmi les positifs et négatifs. Il existe d'autres méthodes de ré-échantillonnage pour rééquilibrer qui sont présentées dans la section suivante.

15.2 Stratégies pour données déséquilibrées

Le problème du déséquilibre a été présenté dans le cadre du modèle logistique. Il n'est cependant pas spécifique à ce modèle et on le trouve présent pour la majorité des algorithmes de prévision. Les méthodes générales qui permettent de répondre aux problèmes du déséquilibre consistent le plus souvent à choisir un critère bien approprié au déséquilibre et/ou à utiliser des méthodes de ré-échantillonnage pour rééquilibrer l'échantillon. Nous présentons ces méthodes dans cette partie.
A partir de maintenant, nous nous plaçons dans le cas où la variable Y à valeurs dans $\{0, 1\}$ est déséquilibrée. La modalité 1 correspond aux positifs et la modalité 0 aux négatifs. Pour fixer les idées, nous supposons que la modalité 1 est sous-représentée, c'est-à-dire que $\Pr(Y = 1)$ est beaucoup plus petit que $\Pr(Y = 0)$, ce qui est le cas le plus fréquent. Ainsi, en épidémiologie un positif correspond à un patient atteint par une pathologie, pathologie souvent rare, en marketing cela pourrait être un client qui a acheté un produit, dans le domaine du crédit bancaire un mauvais payeur, dans la détection de fraude un fraudeur...

15.2.1 Quelques méthodes de rééquilibrage

Le rééquilibrage s'effectue le plus souvent en supprimant des observations de la classe majoritaire (sous-échantillonnage ou *undersampling*) ou en créant de nouvelles observations pour la classe minoritaire (sur-échantillonnage ou *oversampling*). On ne peut bien entendu pas supprimer ou créer des observations n'importe comment. Il existe plusieurs stratégies, nous choisissons d'en proposer quelques-unes qui sont implémentées dans le package **UBL** (*Utility Based Learning*). Elles seront illustrées sur le jeu de données représenté sur la figure 15.5.

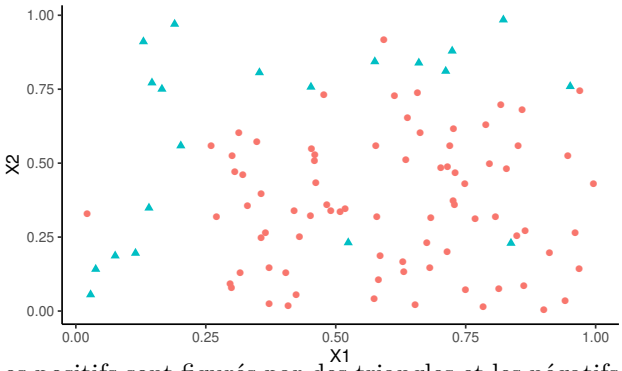


Fig. 15.5 – Les positifs sont figurés par des triangles et les négatifs par des ronds.

Ce jeu contient deux variables explicatives X_1 et X_2 à valeurs dans $[0, 1]$. La modalité 1 (sous-représentée) de Y se situe essentiellement dans deux bandes du carré $[0, 1]^2$: à gauche ($x_1 \leq 0.25$) et en haut ($x_2 \geq 0.75$). Le reste du carré contient essentiellement des 0. On a 20 observations de la classe 1 et 80 de la classe 0.

Sur-échantillonnage ou oversampling

Les méthodes de sur-échantillonnage proposent de créer de nouvelles observations de la classe minoritaire pour rééquilibrer les données. La méthode la plus naïve, appelée **random oversampling**, consiste à dupliquer aléatoirement des observations de cette classe. Elle est très simple à mettre en œuvre, il suffit d'indiquer le nombre de nouvelles observations que l'on souhaite créer. La fonction **RandOverClassifUBL** du package permet d'effectuer un tel ré-échantillonnage.

```
> library(UBL)
> over1 <- RandOverClassif(Y~., dat=df)
> over2 <- RandOverClassif(Y~., dat=df, C.perc=list("0"=1,"1"=2))
> summary(over1$Y)
 0  1
80 80
> summary(over2$Y)
 0  1
80 40
```

On trouvera dans `over1` et `over2` les nouvelles données. Elles sont constituées des observations initiales auxquelles on a ajouté les observations dupliquées. Par défaut, le rééquilibrage est parfait : on duplique aléatoirement de manière à avoir les mêmes nombres de 0 et de 1. L'argument `C.perc` permet de contrôler le niveau de rééquilibrage en indiquant le coefficient multiplicatif que l'on souhaite appliquer aux proportions initiales. Par exemple dans `over2` on conserve tous les 0 et on multiplie par 2 le nombre de 1.
Le random oversampling est facile à mettre en œuvre mais possède l'inconvénient de diminuer la variabilité dans les données. Il existe d'autres algorithmes qui créent de nouvelles observations de la classe minoritaire sans se contenter de les dupliquer. Un des plus utilisés est la méthode SMOTE : *Synthetic Minority Over-sampling TEchnique* (Chawla *et al.*, 2002). Cet algorithme revient à générer des observations entre les segments reliant les individus de la classe minoritaire à certains de leurs plus proches voisins de la même classe (voir algorithme 6).

Algorithme 6 SMOTE

Entrées :
- $(x_i, y_i), i = 1, \ldots, n$ les données ;
- $m \in \{1, \ldots, n\}$ tel que y_m est dans la classe minoritaire ;
- k un entier plus petit que le nombre d'individus de la classe minoritaire.

1. Calculer les k plus proches voisins de x_m parmi les $x_i, i = 1, \ldots, n$ privés de x_m qui sont dans la même classe que x_m ;
2. Choisir au hasard un des plus proches voisins calculés précédemment, on le note $x_{(m)}$;
3. Générer aléatoirement un nouveau point x sur le segment reliant x_m à $x_{(m)}$.

Sortie : la nouvelle observation x à laquelle on associe le label minoritaire.

Le paramètre k doit bien entendu être choisi par l'utilisateur. Il sera forcément lié au nombre d'observations que l'on a dans la classe minoritaire. Une petite valeur de k produit de petits segments, les nouveaux points sont donc proches des points existants et inversement pour une grande valeur de k. Sur R, on peut utiliser la fonction **SmoteClassif** de **UBL** :

```
> set.seed(1234)
> smote1 <- SmoteClassif(Y~., dat=df, k=4)
> smote2 <- SmoteClassif(Y~.,dat=df,k=4, C.perc=list("0"=1,"1"=2))
> summary(smote1$Y)
 0  1
50 50
> summary(smote2$Y)
 0  1
80 40
```

Ici encore le paramètre C.perc permet de contrôler le niveau de ré-équilibrage. Par défaut, il est parfait (autant de 0 que de 1 et chaque niveau possède $n/2$ observations). C'est-à-dire que pour `smote1` :

— on génère 30 nouvelles observations de la classe minoritaire en appliquant l'algorithme 6 aux 20 individus de la classe 1 puis en le ré-appliquant à 10 individus choisis au hasard ;

— on diminue la classe majoritaire en supprimant aléatoirement 30 individus parmi les 80.

Dans le second cas (`smote2`), 20 nouvelles observations de la classe minoritaire sont générées en appliquant l'algorithme 6 aux 20 individus de cette classe, la classe majoritaire reste inchangée. La figure 15.6 représente les deux nouveaux échantillons obtenus.

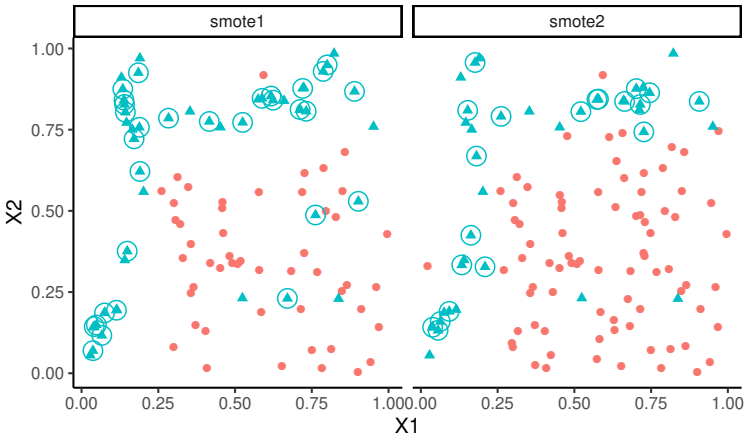


Fig. 15.6 – Oversampling par smote : les nouvelles observations sont entourées.

Sous-échantillonnage ou undersampling

Contrairement aux méthodes précédentes, les algorithmes de sous-échantillonnage suppriment des observations de la classe majoritaire. L'équivalent du random oversampling est le **random undersampling** qui élimine aléatoirement des individus de la classe majoritaire. Cette approche est simple à mettre en œuvre, il suffit d'indiquer les proportions souhaitées pour chaque classe de l'échantillon final. On peut par exemple les spécifier dans l'argument `C.perc` de la fonction **RandUnderClassif** :

```
> under1 <- RandUnderClassif(Y~., dat=df)
> under2 <- RandUnderClassif(Y~., dat=df,
                              C.perc=list("0"=0.5,"1"=1))
> summary(under1$Y)
 0  1
20 20
> summary(under2$Y)
 0  1
40 20
```

Par défaut on supprime autant d'individus que nécessaire pour atteindre l'équilibre parfait. Dans le second cas, on se contente de supprimer la moitié des 0.

Une autre manière de procéder est de repérer les observations de la classe majoritaire qui se situent proches de zones composées d'observations minoritaires. L'algorithme de Tomek (Tomek, 1976) identifie de tels individus à partir de la notion de *lien de Tomek* (Tomek link).

Définition 15.1
Soit $(x_1, y_1), \ldots, (x_n, y_n)$ *n observations à valeurs dans* $\mathbb{R}^p \times \{0, 1\}$. *Une paire* $(x_i, x_j), 1 \leq i \neq j \leq n$ *telle que* $y_i = 0$ *et* $y_j = 1$ *est un T-link si pour tout k différent de i et j on a*

$$\|x_i - x_j\| \leq \|x_i - x_k\| \quad et \quad \|x_i - x_j\| \leq \|x_j - x_k\|.$$

Un T-link est ainsi constitué de deux observations, une de chaque groupe, qui sont plus proches voisins entre elles : on parle de plus proches voisins mutuels. Rééquilibrer par l'algorithme de Tomek revient à identifier les paires qui ont un T-link et à supprimer la paire entière ou uniquement l'individu de la classe majoritaire de la paire. On utilise la fonction **TomekClassif** :

```
> tomek1 <- TomekClassif(Y~.,dat=df)
> tomek2 <- TomekClassif(Y~.,dat=df,rem="maj")
```

Les index des individus supprimés se trouvent dans

```
> tomek1[[2]]
[1]   1   7  12  69  14 100  16  17
> tomek2[[2]]
[1]   1  69 100  16
```

Par défaut l'ensemble des paires qui sont des T-link sont supprimées. L'option `rem="maj"` permet de ne supprimer que les individus de la classe majoritaire qui ont un T-link. Les points supprimés par les algorithmes sont représentés sur la figure 15.7. Les échantillons obtenus, c'est-à-dire le nuage initial sans les points entourés, sont sur la figure 15.8.

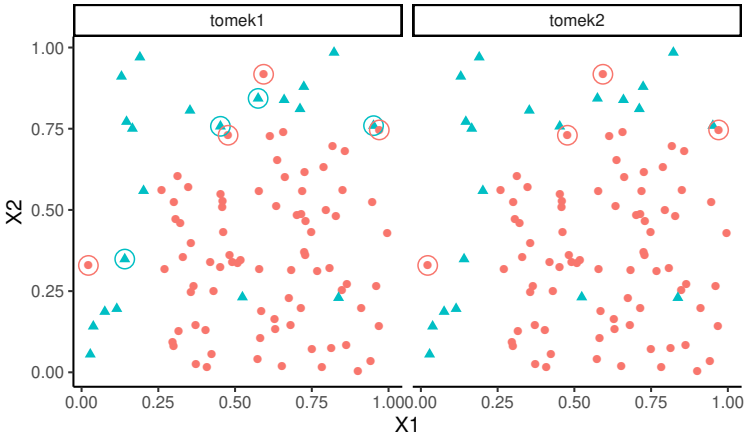


Fig. 15.7 – Oversampling par Tomek : les observations supprimées sont entourées.

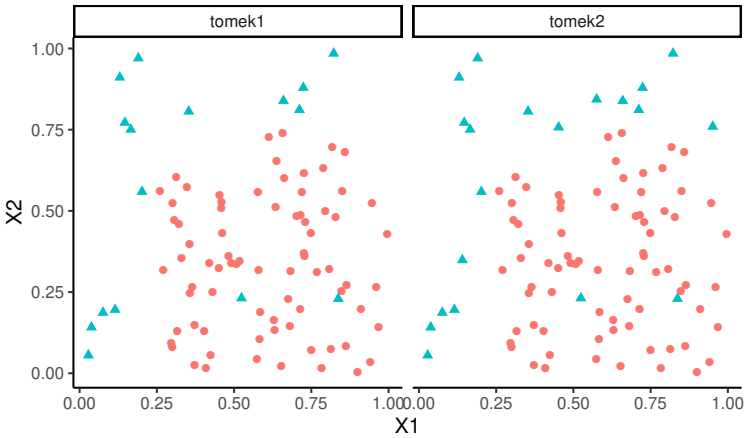


Fig. 15.8 – Oversampling par Tomek : échantillons obtenus.

Nous remarquons que, contrairement aux autres approches, cette méthode ne permet pas de contrôler les degrés de rééquilibrage. Il est par exemple impossible de demander à l'algorithme d'avoir autant de 0 que de 1. Elle va néanmoins permettre d'identifier des individus d'un groupe isolés autour d'individus de l'autre groupe, et donc de clarifier les frontières entre les deux classes.

Remarque

— Les algorithmes SMOTE et Tomek reposent sur des calculs de plus proches voisins qui nécessitent de définir une distance. Sur l'exemple que nous avons utilisé, les données étant dans $\mathbb{R}^2$, nous avons utilisé la distance euclidienne. Il est bien entendu possible d'utiliser d'autres distances, y compris des distances spécifiques à des variables qualitatives. Le choix de la distance peut s'effectuer à l'aide de l'argument `dist` des fonctions présentées.

— L'ensemble des méthodes de rééquilibrage présentées est loin d'être exhaustif. Il existe d'autres algorithmes comme CNN (*Condensed Nearest Neighbors*) ou ENN (*Edited Nearest Neighbors*)... Ces méthodes sont également implémentées dans le package **UBL**.

15.2.2 Critères pour données déséquilibrées

Les techniques de rééquilibrage présentées dans la section précédente permettent de construire un nouvel échantillon (un par technique) sur lequel on va estimer un (ou plusieurs) modèle(s). Chaque technique va ainsi donner un (ou plusieurs) modèle(s) estimé(s) qu'il est maintenant nécessaire de comparer. Toutes les techniques de comparaison entre différents modèles vues au chapitre précédent (p. 321) peuvent donc être utilisées ici. Les critères pour les scores tels que les courbes ROC ou lift ne nécessitent pas d'ajustement particulier en présence de données déséquilibrées. Par contre les critères mesurant la performance de règles de classification (voir section 14.2, p. 323) comme l'*accuracy* ne sont pas forcément adaptés ici et il existe des critères spécifiques les complétant (ou les remplaçant). Nous les présentons ici.

On rappelle qu'une règle de prévision est représentée par une fonction $g : \mathbb{R}^p \to \{0, 1\}$ qui renverra la prévision $g(x)$ pour une nouvelle observation $x \in \mathbb{R}^p$. Les critères sont généralement calculés à partir d'une table de confusion qui confronte les prévisions faites par l'algorithme g aux valeurs observées de Y :

Vrai \ Prévu	$g(X) = 0$	$g(X) = 1$
$Y = 0$	VN	FP
$Y = 1$	FN	VP

Tableau 15.2 – Table de confusion.

Le terme VN désigne ici le nombre de vrais négatifs, c'est-à-dire le nombre d'individus pour lesquels la règle a prédit 0 et où on a effectivement observé 0. Cette table, qui s'obtient le plus souvent à l'aide de méthodes de ré-échantillonnage de type validation croisée (voir section 14.2), permet de visualiser la capacité à bien prédire de la règle g dans les populations positives et négatives.

Afin d'illustrer et comparer les différents critères, nous considérons l'exemple suivant où, pour un même jeu de données, on dispose de deux algorithmes de prévision donnant lieu à deux vecteurs de prévision `P1` et `P2` (les données à prévoir sont dans `Y`). Les tables de contingence sont présentées ici :

```
> table(Y,P1)
   P1
Y     0   1
  0 468   0
  1  31   1
> table(Y,P2)
   P2
Y     0   1
  0 407  61
  1   4  28
```

Ces données contiennent 32 positifs contre 468 négatifs. On remarque que l'algorithme 1 a tendance à quasiment toujours prédire la classe sur-représentée tandis que le 2 parvient à identifier une partie plus importante des individus de la classe sous-représentée (28 sur 32). L'algorithme 2 peut donc être préférable au 1 si l'objectif est d'identifier des positifs.
Nous commençons par étudier un des critères les plus couramment utilisés : l'*accuracy*. On rappelle qu'il est défini par

$$\text{Acc} = \Pr(g(X) = Y)$$

et estimé selon

$$\widehat{\text{Acc}} = \frac{\text{VN} + \text{VP}}{n}.$$

Pour l'exemple considéré, il est facile de voir que le premier algorithme est privilégié par l'*accuracy* puisqu'il réalise 469 bonnes prévisions contre 435 pour le second. Cela ne signifie pas pour autant que l'algorithme 1 est meilleur puisqu'il ne détecte qu'un seul positif. En effet, l'*accuracy* n'est pas un bon critère pour mesurer la capacité à bien identifier la population sous-représentée. On peut l'expliquer assez facilement en remarquant que

$$\text{Acc} = \Pr(g(X) = 1|Y = 1)\Pr(Y = 1) + \Pr(g(X) = 0|Y = 0)\Pr(Y = 0).$$

Ce critère s'obtient donc en pondérant la sensibilité $\Pr(g(X) = 1|Y = 1)$ et la spécificité $\Pr(g(X) = 0|Y = 0)$ par les probabilités a priori $\Pr(Y = 1)$ et $\Pr(Y = 0)$. Ainsi dans le cas où les négatifs sont sur-représentés ($\Pr(Y = 0)$ grand), ce critère a tendance à privilégier la spécificité et donc des règles qui identifient bien les négatifs. La règle $g(x) = 0\,\forall x \in \mathbb{R}^p$ peut par exemple se révéler très performante dans ce cas-là pour l'*accuracy* alors qu'elle ne permet pas d'identifier le moindre positif. Une manière assez évidente de corriger l'*accuracy* est de donner le même poids à la sensibilité et à la spécificité. On obtient alors la **balanced accuracy** :

$$\text{Balanced Acc} = \frac{\Pr(g(X) = 1|Y = 1) + \Pr(g(X) = 0|Y = 0)}{2}.$$

Ce critère varie entre 0 (règle qui prédit l'inverse de ce qui est observé) et 1 (prévisions parfaites). Nous remarquons que le poids de la classe sur-représentée n'est pas pris en compte dans le critère, d'où l'intérêt de l'utiliser lorsque les données sont déséquilibrées. Sur l'exemple, la *balanced accuracy* vaut 0.516 pour l'algorithme 1 et 0.872 pour l'algorithme 2 qui est donc privilégié.

Remarque
La *balanced accuracy* donne le même poids à la sensibilité et à la spécificité. Dans certains cas, on dispose d'une information auxiliaire qui permet de quantifier l'importance de ces deux critères. Si on sait par exemple qu'il est 5 fois plus important de détecter les positifs que les négatifs, il est alors facile de prendre en considération cette information en définissant un nouveau critère qui prendra en compte ces poids, par exemple

$$\frac{5\Pr(g(X)=1|Y=1)+\Pr(g(X)=0|Y=0)}{6}.$$

Dans le même ordre d'idée que la *balanced accuracy* qui est la moyenne arithmétique entre la sensibilité et la spécificité, le F_1**-score** est défini comme la moyenne harmonique entre la *précision*, qui mesure la capacité à bien détecter les positifs parmi les positifs prédits, et le *recall* (ou spécificité) qui mesure la capacité à bien prédire les positifs :

$$F_1 = 2\,\frac{\text{Precision } \times \text{Recall}}{\text{Precision } + \text{Recall}},$$

où

$$\text{Precision} = \Pr(Y=1|g(X)=1) \quad \text{et} \quad \text{Recall} = \Pr(g(X)=1|Y=1).$$

Ici encore, ce critère ne prend pas en compte les tailles des classes et est donc moins sensible à un problème de données déséquilibrées. Le F_1-score se concentre sur la capacité à bien identifier les positifs (*Recall*), en évitant de trop en prédire (Precision). Il varie entre 0 (mauvaise identification des positifs) et 1 (identification parfaite). Sur l'exemple, on a un F_1-score de 0.060 pour l'algorithme 1 contre 0.463 pour le second qui est donc privilégié. Il est important de remarquer que, contrairement à l'*accuracy* et à la *balanced accuracy*, le F_1-score n'est pas symétrique vis-à-vis du choix que l'on a fait pour définir les positifs. En effet, si on inverse les 1 et les 0, la valeur du F_1-score change. Il est donc crucial de bien définir la classe des positifs en fonction de ce que l'on souhaite évaluer.
La *balanced accuracy* et le F_1-score sont définis à partir de moyennes entre des probabilités qui mesurent le capacité à bien détecter les positifs et/ou négatifs. Le **kappa de Cohen** (Cohen, 1960) est défini différemment, il propose de comparer

— la probabilité de bien classer (accord), c'est-à-dire l'*accuracy* que l'on note ici $\Pr(a)$;
— à une probabilité de bien classer de façon **aléatoire** que l'on note $\Pr(e)$. Le terme aléatoire est à prendre dans le sens où on suppose que la prévision $g(X)$

est indépendante de l'observation Y :

$$\begin{aligned}\Pr(e) &= \Pr_{\text{al}}(g(X) = Y)\\ &= \Pr(g(X) = 0)\Pr(Y = 0) + \Pr(g(X) = 1)\Pr(Y = 1).\end{aligned}$$

Le κ **de Cohen** est alors défini par

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}.$$

Cette définition peut paraître étrange et il convient de l'expliquer afin de comprendre en quoi ce critère peut être intéressant pour des données déséquilibrées. Remarquons tout d'abord que si g est parfaite, c'est-à-dire $g(X) = Y$, alors l'*accuracy* $\Pr(a)$ vaut 1 tout comme le kappa de Cohen et on ne pourra pas faire mieux. Plaçons-nous maintenant dans un cas de données (très) déséquilibrées, c'est-à-dire avec $\Pr(Y = 0)$ grand et considérons la règle qui prédit toujours 0, cette règle n'identifie donc pas les positifs. On a alors

$$\Pr(a) = \Pr(Y = 0) \quad \text{et} \quad \Pr(e) = \Pr(Y = 0).$$

Dans ce cas, l'*accuracy* est élevée (égale à $\Pr(Y = 0)$) tandis que le kappa de Cohen sera nul. D'où l'intérêt d'utiliser ce critère lorsque les données sont déséquilibrées. Sur l'exemple de cette partie, le kappa de Cohen vaut 0.057 pour l'algorithme 1 contre 0.407 pour l'algorithme 2 qui est donc privilégié pour ce critère.
Les critères présentés dans cette partie peuvent être directement estimés à partir d'une table de contingence ou de deux vecteurs qui contiennent les valeurs prédites et observées. Nous proposons d'utiliser le package `yardstick` qui possède des fonctions spécifiques pour calculer un très grand nombre de critères. On peut par exemple obtenir l'*accuracy*, la *balanced accuracy*, le F_1-score et le kappa de Cohen avec les fonctions **accuracy**, **bal_accuracy**, **f_meas** et **kap**. Nous les calculons ici pour le vecteur de prévision `P2` :

```
> library(yardstick)
> df <- data.frame(Y,P2)
> multi_metric <- metric_set(accuracy,bal_accuracy,f_meas,kap)
> multi_metric(df,truth=Y,estimate=P2,event_level="second")
# A tibble: 4 x 3
  .metric      .estimator .estimate
  <chr>        <chr>          <dbl>
1 accuracy     binary         0.87
2 bal_accuracy binary         0.872
3 f_meas       binary         0.463
4 kap          binary         0.407
```

Nous avons utilisé l'option `event_level="second"` pour préciser que la classe des positifs correspond au second niveau de `Y`. Nous avons vu que ce choix est important pour certains critères comme le F_1-score.

15.3 Choisir un algorithme de rééquilibrage

La section précédente a proposé des algorithmes qui permettent de rééquilibrer les échantillons ainsi que des critères adaptés aux données déséquilibrées. Bien entendu, la vraie question pour le statisticien est de trouver le « meilleur » algorithme permettant de répondre à son problème avec un échantillon présentant des données déséquilibrées. Il doit donc faire des choix entre les différentes stratégies possibles. Ces choix sont difficiles car prendre en compte des données déséquilibrées n'est qu'une étape parmi d'autres dans un projet de sciences des données ; citons par exemple

— la gestion des données manquantes ;
— le choix de variables : faut-il enlever ou non des variables qui apporteraient une même information ?
— le choix d'un algorithme de rééquilibrage, il en existe plein et de plus ces algorithmes dépendent de paramètres qu'il faut calibrer...
— le choix d'un algorithme de classification (logistique, ridge, lasso...) et d'une manière de choisir les paramètres de cet algorithme (validation croisée...).
— ...

Cette partie n'a bien évidemment pas vocation à donner un protocole précis pour mener à bien un projet de data-science car un tel protocole n'existe pas. De plus, pour réussir son projet il est nécessaire de bien connaître les données et l'objectif, ce qui change à chaque projet. Néanmoins, pour atteindre cet objectif il est souvent nécessaire de faire des essais, c'est-à-dire comparer des procédures. Nous proposons donc ici de donner des pistes pour le choix de l'algorithme de rééquilibrage.

15.3.1 Rééquilibrage et validation croisée

Etant donné $\mathcal{D}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ où les (x_i, y_i) sont à valeurs dans $\mathbb{R}^p \times \{0, 1\}$ on rappelle qu'un algorithme de prévision est représenté par une fonction g

$$g : \mathbb{R}^p \times (\mathbb{R}^p \times \{0, 1\})^n \to \mathbb{R} \tag{15.4}$$

qui à un nouvel $x \in \mathbb{R}^p$ va renvoyer la prévision $g(x, \mathcal{D}_n)$. La notation est un peu lourde mais elle permet de visualiser que cette prévision est calculée à partir des données $\mathcal{D}_n$. Cette prévision peut concerner le groupe (0 ou 1) ou le score (un réel). Nous insistons sur le fait que l'algorithme g doit inclure toutes les étapes présentées précédemment (données manquantes, rééquilibrage, régression logistique, choix de variables...). Prenons l'exemple où on souhaite combiner le "random oversampling" avec la régression logistique, la prévision $g(x, \mathcal{D}_n)$ est alors obtenue

1. en appliquant le random oversampling sur $\mathcal{D}_n$. On va donc obtenir un nouvel échantillon rééquilibré $(\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_m, \tilde{y}_m)$ pas forcément de taille n ;

2. en estimant les paramètres du modèle logistique

$$\mathsf{logit}\, p_\beta(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

avec l'échantillon rééquilibré.

Le score prédit sera alors

$$g(x, \mathcal{D}_n) = \frac{\exp(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p)}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p)}.$$

Il est ainsi possible (et facile) de construire un grand nombre d'algorithmes. Pour les comparer, on utilise généralement des critères comme ceux présentés dans la section 15.2.2 (*accuracy*, *balanced accuracy*, kappa de Cohen, AUC...). Bien entendu le choix du critère est capital et dépend des objectifs métiers (importance de détecter les positifs pour le problème considéré par exemple). Il est généralement recommandé d'utiliser plusieurs critères afin de visualiser la robustesse des algorithmes vis-à-vis de ces critères. Mathématiquement ces critères sont toujours calculés à partir de la confrontation des valeurs observées et prédites par l'algorithme. Un critère ou risque peut donc être représenté par une fonction

$$\begin{aligned} \mathcal{R} : \mathbb{R}^n \times \mathbb{R}^n &\to \mathbb{R} \\ (y_1^n, \hat{y}_1^n) &\mapsto \mathcal{R}(y_1^n, \hat{y}_1^n) \end{aligned} \tag{15.5}$$

où $y_1^n = (y_1, \ldots, y_n)$ désigne le vecteur des valeurs observées et $\hat{y}_1^n = (\hat{y}_1, \ldots, \hat{y}_n)$ celui des valeurs prédites. Comme nous l'avons vu dans les chapitres 10.1 et 14, les prévisions doivent se calculer à l'aide de méthodes de ré-échantillonnage. Nous rappelons l'algorithme de validation croisée dans ce contexte.

Algorithme 7 Validation croisée pour données déséquilibrées

Entrées :
- $\mathcal{D}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ les données ;
- $\{\mathcal{I}_1, \ldots, \mathcal{I}_K\}$ une partition de $\{1, \ldots, n\}$ en K blocs ;
- g un algorithme de prévision comme dans (15.4) ;
- $\mathcal{R}$ un risque comme dans (15.5).

Pour $k = 1, \ldots, K$

1. Poser $\mathcal{I}_{\text{train}} = \{1, \ldots, n\} \backslash \mathcal{I}_k$ et $\mathcal{I}_{\text{test}} = \mathcal{I}_k$.
2. Construire l'algorithme de prévision sur $\mathcal{D}_{\text{train}} = \{(x_i, y_i) : i \in \mathcal{I}_{\text{train}}\}$, on le note $g_k(.) = g(., \mathcal{D}_{\text{train}})$.
3. En déduire $\hat{y}_i = g_k(x_i)$ pour chaque $i \in \mathcal{I}_{\text{test}}$.

Sortie : $\mathcal{R}(y_1^n, \hat{y}_1^n)$.

Plaçons-nous dans le cas où l'algorithme comporte une étape de rééquilibrage. Nous remarquons qu'à chaque itération cette étape de rééquilibrage ne doit être appliquée qu'à l'échantillon d'apprentissage $\mathcal{D}_{\text{train}}$, pas à l'échantillon test. Ceci est logique dans la mesure où, en pratique, l'algorithme choisi sera appliqué sur des nouvelles données qui ne seront pas rééquilibrées. C'est pourquoi il est important de ne pas rééquilibrer les échantillons tests lorsqu'on cherche à évaluer la performance d'un algorithme.

15.3.2 Application aux données d'images publicitaires

On reprend les données qui ont été présentées dans la section 14.6. Nous avons supprimé les 4 variables qui contenaient des données manquantes, on se retrouve en présence de 3 279 individus et 1 554 variables explicatives. On fait un résumé de la variable à expliquer

```
> summary(ad.data1$Y)
   ad. nonad.
   459   2820
```

et on remarque qu'elle comporte 14 % d'individus qui correspondent à des publicités. Afin d'utiliser les mêmes notations que précédemment, on recode les modalités de Y en 0-1 :

```
> library(tidyverse)
> ad.data1 <- ad.data1 %>%
+   transform(Y=fct_recode(Y, "0"="nonad.", "1"="ad.")) %>%
+   transform(Y=fct_inseq(Y))
```

On propose de comparer des algorithmes de prévision issus d'une régression logistique avec maximum de vraisemblance, ridge et lasso combinés avec les 4 méthodes de rééquilibrage présentées dans la section 15.2.1 : random oversampling, smote, random undersampling et Tomek. De plus, afin d'évaluer l'éventuelle pertinence du rééquilibrage, nous construirons également les algorithmes sur les données brutes (sans rééquilibrage). On propose donc de comparer 15 algorithmes de prévision.
Nous allons effectuer la comparaison à l'aide d'une validation croisée 10 blocs pour déduire les estimations des critères présentés dans la section 15.2.2. On définit tout d'abord les 10 blocs de la validation croisée :

```
> set.seed(1234)
> bloc <- sample(1:10, nrow(ad.data1), replace=TRUE)
> table(bloc)
```

Puis on définit une liste SCORE de longueur 5 (autant que d'algorithmes de rééquilibrage) dont chaque élément contient un `dataframe` avec les probabilités que l'image soit une publicité estimée par chaque méthode (3 méthodes, donc 3 colonnes) :

```
> score <- data.frame(matrix(0, nrow=nrow(ad.data1), ncol=3))
> names(score) <- c("logit", "lasso", "ridge")
> SCORE <- list(brute=score, over=score, smote=score,
+               under=score, tomek=score)
```

On peut maintenant faire la validation croisée. Nous allons faire une boucle sur `k` et pour chaque valeur de `k` entre 1 et 10 :

```
> for (k in 1:10) {
```

1. on détermine les échantillons d'apprentissage et de test :

```
+  ind.test <- bloc==k
+  dapp <- ad.data1[!ind.test, ]
+  dtest <- ad.data1[ind.test, ]
+  X.test <- model.matrix(Y~., data=dtest)[, -1]
```

2. on rééquilibre les données d'apprentissage uniquement :

```
+  ech.app <- list(norm=dapp,
+                  over=RandOverClassif(Y~., dat=dapp),
+                  smote=SmoteClassif(Y~., dat=dapp),
+                  under=RandUnderClassif(Y~., dat=dapp),
+                  tomek=TomekClassif(Y~., dat=dapp)[[1]])
+  mod.mat.list <- function(df){ model.matrix(Y~., data=df)[,-1] }
+  Y.list <- function(df) df$Y
+  X.app <- lapply(ech.app, mod.mat.list)
+  Y.app <- lapply(ech.app, Y.list)
```

Les 5 types d'échantillons (un par méthode de rééquilibrage) sont regroupés dans 2 listes `X.app` et `Y.app` contenant les variables explicatives et à expliquer respectivement ;

3. on entraîne les modèles logistique, lasso et ridge sur tous les échantillons d'apprentissage. Le paramètre de régularisation des méthode lasso et ridge est sélectionné en faisant une validation croisée 10 blocs, c'est-à-dire que l'échantillon d'apprentissage est découpé en 10 blocs pour choisir ce paramètre :

```
+  for (j in 1:5){
+    lasso <- cv.glmnet(X.app[[j]], Y.app[[j]], family="binomial")
+    ridge <- cv.glmnet(X.app[[j]], Y.app[[j]], family="binomial",
+                       alpha=0)
+    logit <- glm(Y~., data=ech.app[[j]], family="binomial")
```

4. on calcule enfin les probabilités estimées que l'image soit une publicité pour chaque individu de l'échantillon test :

```
+    SCORE[[j]][ind.test,] <- data.frame(
+     logit=predict.glm(logit, newdata=dtest, type="response"),
+     lasso=as.vector(predict(lasso,newx=X.test,type="response")),
+     ridge=as.vector(predict(ridge,newx=X.test,type="response")))
+  } # fin de boucle j
```

```
+ } # fin de boucle k
```

On assemble tous les résultats dans un seul data-frame de 4 colonnes qui contiendra :
— le nom de l'algorithme de rééquilibrage (colonne `meth`) ;
— les valeurs observées de Y pour chaque individu (0 ou 1, colonne `obs`) ;
— l'algorithme de prévision (logistique, lasso, ridge, colonne `algo`) ;
— la probabilité que l'image soit une publicité estimée par l'algorithme (colonne `score`).

```
> mat.score <- bind_rows(brutes=SCORE[[1]],
+                        over=SCORE[[2]],
+                        smote=SCORE[[3]],
+                        under=SCORE[[4]],
+                        tomek=SCORE[[5]], .id="meth") %>%
+   mutate(obs=rep(ad.data1$Y, 5))  %>%
+   pivot_longer(c(logit, lasso, ridge),
+                names_to="algo", values_to="score")
```

On déduit de cette matrice tous les critères de performance présentés dans la section 15.2.2. On obtient par exemple les AUC du tableau 15.3 avec

```
> library(yardstick) #pour la fonction roc_auc
> mat.score %>% group_by(meth,algo) %>%
+     roc_auc(truth=obs,estimate=score,event_level="second") %>%
+     pivot_wider(-c(.metric,.estimator),names_from=algo,
+                 values_from=.estimate)
```

meth	lasso	logit	ridge
brutes	0.943	0.831	**0.980**
over	0.973	0.790	0.977
smote	0.973	0.680	**0.977**
tomek	0.950	0.763	**0.979**
under	0.956	0.787	0.964

Tableau 15.3 – AUC.

La plupart des autres critères (*accuracy*, kappa de Cohen...) nécessitent d'avoir les groupes prédits par les algorithmes. Ces prévisions s'obtiennent en fixant un seuil s entre 0 et 1 et en prédisant « groupe 1 » si la probabilité est plus grande que s, 0 sinon. On peut utiliser les méthodes de choix de seuil présentées dans la section 14.4. Par souci de concision, nous présentons les résultats pour le seuil de 0.5. On ajoute une nouvelle colonne `prev` qui contiendra les groupes prédits :

```
mat.score <- mat.score %>% mutate(prev=as.factor(round(score)))
```

On trouvera les nouveaux critères obtenus à partir de ces prévisions dans les tableaux 15.4-15.7. Les trois meilleures valeurs sont affichées en gras. Ces tableaux s'obtiennent sur R en adaptant le code précédent. Il faudra par exemple remplacer `roc_auc` par `kap` et `estimate=score` par `estimate=prev` pour calculer le tableau 15.7.

meth	lasso	logit	ridge
brutes	0.969	0.886	**0.970**
over	0.961	0.847	0.963
smote	0.960	0.699	0.960
tomek	**0.969**	0.818	**0.970**
under	0.954	0.808	0.955

Tableau 15.4 – Accuracy.

meth	lasso	logit	ridge
brutes	0.898	0.832	0.900
over	0.931	0.806	**0.935**
smote	**0.933**	0.680	**0.933**
tomek	0.896	0.797	0.900
under	0.921	0.792	0.899

Tableau 15.5 – *balanced accuracy*.

meth	lasso	logit	ridge
brutes	**0.879**	0.650	**0.884**
over	0.864	0.579	0.873
smote	0.863	0.378	0.862
tomek	0.876	0.542	**0.882**
under	0.843	0.529	0.837

Tableau 15.6 – F_1 score.

meth	lasso	logit	ridge
brutes	**0.862**	0.584	**0.867**
over	0.842	0.491	0.851
smote	0.839	0.224	0.838
tomek	0.859	0.440	**0.866**
under	0.816	0.422	0.811

Tableau 15.7 – Kappa de Cohen.

Il n'est pas facile de synthétiser tous ces résultats. Plusieurs tendances semblent néanmoins se dégager :

— le modèle logistique est clairement moins performant que les algorithmes ridge et lasso. Nous l'avions déjà remarqué dans la section 14.6. Cela s'explique par le nombre important de variables dans le modèle. Les méthodes régularisées sont plus pertinentes dans ce contexte,

— les performances des régressions ridge et lasso sont proches, avec une préférence pour le ridge,

— l'apport des méthodes de rééquilibrage est discutable sur cet exemple. On remarque en effet que les algorithmes appliqués sur les données brutes (non rééquilibrées) sont performants. Le rééquilibrage peut néanmoins améliorer certains critères comme la *balanced accuracy* pour lequel les méthodes de sur-échantillonnage (random oversampling et smote) permettent d'améliorer.

Nous terminons en présentant quelques résultats pour d'autres valeurs de seuil. En effet, nous avons ici calculé la plupart des critères en utilisant le seuil de 0.5 pour déduire les groupes prédits à partir des valeurs de score. Il est souvent intéressant de visualiser ces critères pour différentes valeurs de seuil, on utilise alors des grilles de score qui consistent à présenter les valeurs de critère en fonction de différentes valeurs de seuil. Nous présentons dans le tableau 15.8 la grille de score de la méthode ridge appliquée sur les données rééquilibrées par l'algorithme smote. Cette grille peut aider l'utilisateur à choisir le seuil de prévision en fonction des objectifs métiers.

seuil	sens	spec	Acc	bal.acc	F1	kappa
0	0.000	1.000	0.140	0.500	0.246	0.000
0.1	0.671	0.955	0.711	0.813	0.480	0.343
0.2	0.745	0.932	0.771	0.839	0.533	0.416
0.3	0.851	0.911	0.860	0.881	0.645	0.566
0.4	0.918	0.885	0.913	0.901	0.740	0.690
0.5	0.983	0.843	0.964	0.913	0.867	0.846
0.6	0.991	0.797	0.964	0.894	0.861	0.841
0.7	0.995	0.739	0.959	0.867	0.834	0.811
0.8	0.997	0.672	0.952	0.835	0.796	0.770
0.9	0.999	0.562	0.938	0.780	0.717	0.685
1	1.000	0.000	0.860	0.500		0.000

Tableau 15.8 – Grille de score pour l'algorithme ridge combiné avec la méthode smote.

15.4 Exercices

Exercice 15.1 (Critères pour un exemple de données déséquilibrées)

1) Générer un vecteur d'observations `Y` de taille 500 selon une loi de Bernoulli de paramètre 0.05.

2) Générer un vecteur de prévisions `P1` de taille 500 selon une loi de Bernoulli de paramètre 0.005.

3) Générer un vecteur de prévision `P2` de taille 500 tel que

$$\mathcal{L}(P2|Y=0) = \mathcal{B}(0.10) \quad \text{et} \quad \mathcal{L}(P2|Y=1) = \mathcal{B}(0.85).$$

4) Calculer les tables de contingence de `P1` et `P2` à l'aide de **table**.

5) Pour `P2`, calculer, avec les fonctions usuelles de R l'*accuracy*, le *recall* et la précision.

6) Même question pour le F_1-score et le kappa de Cohen.

7) Retrouver ces indicateurs à l'aide de la fonction **multi_metric** du package **yardstick** puis comparer les prévisions `P1` et `P2`.

Exercice 15.2 (Echantillonnage rétrospectif)

Démontrer la formule (15.2).

Exercice 15.3 (Rééquilibrage)

Les fichiers `dd_exo3_1.csv`, `dd_exo3_2.csv` et `dd_exo3_3.csv` contiennent chacun $n = 1000$ observations d'une variable binaire Y et de deux variables continues X_1 et X_2. On veut expliquer Y par X_1 et X_2 à chaque fois.

1) Comparer les distributions de Y pour ces trois jeux de données et visualiser les observations.

2) Séparer les trois jeux de données en un échantillon d'apprentissage et un échantillon test. On mettra deux tiers d'observations dans l'échantillon d'apprentissage.

3) Ajuster une régression logistique sur les 3 échantillons d'apprentissage, calculer les labels prédits sur les échantillons test (avec un seuil de 0,5) et estimer les différents indicateurs de la section 15.2.2 à l'aide de la fonction **confusionMatrix**. On prendra en compte l'interaction entre X_1 et X_2 dans le modèle.

4) Rééquilibrer les données `dd_exo3_3.csv` en utilisant des techniques de la section 15.2.1. Visualiser les nouveaux échantillons obtenus.
5) Recalculer les différents critères sur les échantillons rééquilibrés.

Exercice 15.4 (Rééquilibrage et information de Fisher)
Nous nous plaçons dans le cadre d'un modèle logistique où $Y|X = x$ suit une loi exponentielle de densité $f_\theta(y|x)$. Nous supposons l'existence d'une variable binaire S (modélisant le rééquilibrage) dont la loi conditionnelle $S|Y$ ne dépend pas de X et dont sa densité h n'est pas fonction de θ. L'échantillon non rééquilibré est constitué de n observations iid de (Y_i, S_i, X_i) et celui rééquilibré des observations où $S_i = 1$.
1) Vérifier intuitivement que la situation de sous-échantillonnage proposé en page 364 peut être approximée par ce schéma si la taille du sous échantillons des $Y = 0$ (dans lequel on sélectionne) est grande.
2) Notons g la densité conditionnelle du couple $(Y, S|X)$ (dont nous supposons l'existence) ; retrouver que $g(y, s|x) = h(s|y)f_\theta(y|x)$ et déduire que la log-vraisemblance des données augmentées (Y_i, S_i) vaut

$$\mathcal{L}^*(\theta) = \mathcal{L}(\theta) + \sum_{i=1}^{n} \log h(s_i|y_i).$$

où $\mathcal{L}(\theta)$ dénote la log-vraisemblance des données.
3) En déduire en dérivant 2 fois par rapport à θ que

$$I_n^*(\theta) = I_n(\theta).$$

4) En notant $m(s|x)$ la densité de $S|X$ et avec $g(y, s|x) = l(y|s, x)m(s|x)$ montrer que

$$\mathcal{L}^*(\theta) = \sum_{i\in\{i|S_i=1\}} \log l(y|s = 1, x) + \sum_{i\in\{i|S_i=0\}} \log l(y|s = 0, x) + \sum_{i=1}^{n} \log m(s_i|x_i).$$

5) En déduire en dérivant 2 fois par rapport à θ que

$$I_n^*(\theta) = -\frac{\partial^2}{\partial\theta^2} \sum_{i\in\{i|S_i=1\}} \log l(y|s = 1, x) - \frac{\partial^2}{\partial\theta^2} \sum_{i\in\{i|S_i=0\}} \log l(y|s = 0, x) - \frac{\partial^2}{\partial\theta^2} \sum_{i=1}^{n} \log m(s_i|x_i).$$

Notons ces 3 informations de Fisher $I_n(\theta|S = 1)$, $I_n(\theta|S = 0)$ et $I_n(\theta, S)$ respectivement.
6) Parmi toutes les informations de Fisher introduites, $I_n^*(\theta)$, $I_n(\theta)$, $I_n(\theta|S = 1)$, $I_n(\theta|S = 0)$ et $I_n(\theta, S)$,
— quelle est celle qui correspond à l'échantillon ?
— quelle est celle qui correspond à l'échantillon après rééquilibrage ?
7) En supposant toutes les conditions réunies, chaque information de Fisher converge (quand $n \to \infty$) vers une matrice définie positive. Vérifier sans calcul l'assertion suivante : Asymptotiquement, pour l'inférence sur θ, effectuer un rééquilibrage des données selon un tirage binaire indépendant de X conditionnellement à Y apporte moins d'information que de ne pas le faire.

Cinquième partie

Introduction à la régression non paramétrique

Chapitre 16

Introduction à la régression spline

L'objectif de ce chapitre est de présenter de façon simplifiée une introduction aux méthodes de régression non paramétrique, en particulier la régression spline et les estimateurs à noyau et des k plus proches voisins. Le lecteur souhaitant approfondir ses connaissances pourra consulter le livre de Eubank (1999) par exemple.

16.1 Introduction

Nous avons évoqué en section 2.2 (p. 34) de possibles extensions du modèle de régression simple *via* la régression polynomiale, qui peut être considérée comme une régression multiple. Nous allons évoquer les avantages et les inconvénients potentiels de la régression polynomiale en reprenant l'exemple univarié de l'ozone vu au chapitre 1.

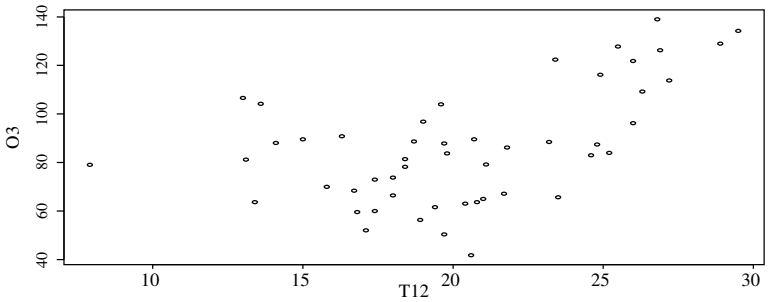


Fig. 16.1 – 50 données journalières de température et O3.

Chaque point du graphique représente, pour un jour donné, une mesure de la température à 12 h et le pic d'ozone de la journée.

Nous avons déjà étudié en détail la régression linéaire simple. Le modèle supposé était

$$y_i \quad = \quad f(x_i) + \varepsilon_i, \quad i = 1, \cdots, n,$$

où la fonction f était de la forme $ax + b$. Cependant, il y a peu de raison pour que la fonction inconnue soit de cette forme et nous allons essayer d'améliorer le modèle en supposant dans un premier temps que la fonction inconnue est de type polynomial. Le problème de minimisation univarié s'écrit toujours :

$$\underset{f \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - f(x_i)\right)^2,$$

mais $\mathcal{G}$ est maintenant l'espace des polynômes de degré d. Nous pouvons donc estimer le paramètre $\beta = (\beta_0, \beta_1, \ldots, \beta_d)$ en minimisant la quantité

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i - \cdots - \beta_d x_i^d\right)^2.$$

La matrice du plan d'expérience est $X = (1, x, x^2, \cdots, x^d)$ et tous les résultats vus au chapitre 2 restent valables. L'avantage des polynômes est la facilité de mise en œuvre ainsi que leur capacité d'approximation. En effet, si la fonction inconnue f à estimer est continue sur un compact, elle peut être approchée uniformément par des polynômes sur ce compact (théorème de Stone-Weierstrass). La preuve du théorème de Stone-Weierstrass montre que le degré du polynôme croît avec la qualité de l'approximation.
Il est donc intéressant d'étudier le comportement de l'estimateur provenant de la régression polynomiale en fonction du degré d du polynôme. Si d est petit, la fonction sous-jacente sera peu flexible (imaginons le cas d'une droite ou d'une parabole), alors que si d est élevé (imaginons un polynôme de degré 9 par exemple) la fonction aura tendance à osciller fortement.
La fonction R **polyreg** à faire en exercice 16.2 effectue une régression polynomiale de degré d. Sur le graphique suivant sont représentées les courbes obtenues par les moindres carrés pour les modèles polynomiaux d'ordre 1, 2, 3 et 9 *via* les commandes suivantes :

```
> plot(O3 ~ T12, xlim=c(0,35), ylim=c(0,150), data=ozone)
> iter <- 1
> for(ii in c(1,2,3,9)){
+ tmp <- polyreg(ozone,d=ii)
+ lines(tmp$grillex,tmp$grilley,col=iter,lty=iter)
+ iter <- iter+1
+ }
> legend(15,150,c("d=1","d=2","d=3","d=9"),col=1:4,lty=1:4)
```

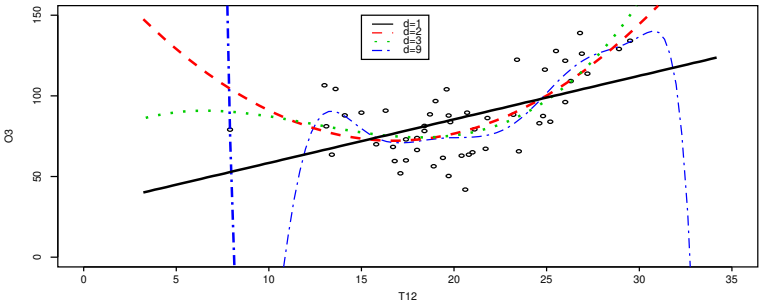


Fig. 16.2 – Différentes régressions polynomiales pour les données d'ozone.

De façon générale, un polynôme de degré d peut changer de monotonie au maximum $d-1$ fois. Si le degré d est trop élevé, la courbe estimée risque d'osciller exagérément. Cependant, lors de la régression polynomiale, même si la fonction estimée admet peu de variations, il faut faire attention aux valeurs que peut prendre la fonction entre les observations. Cela est flagrant (figure 16.2) pour la régression polynomiale de degré 9 dans l'intervalle $[7, 10]$. Ce propos est encore plus vrai en dehors de l'intervalle des données où il est proscrit d'effectuer des prévisions.
Un autre problème de la régression polynomiale est son caractère global. En effet, une modification d'un point de l'échantillon peut modifier complètement la courbe estimée, surtout si ce point est au bord du domaine.

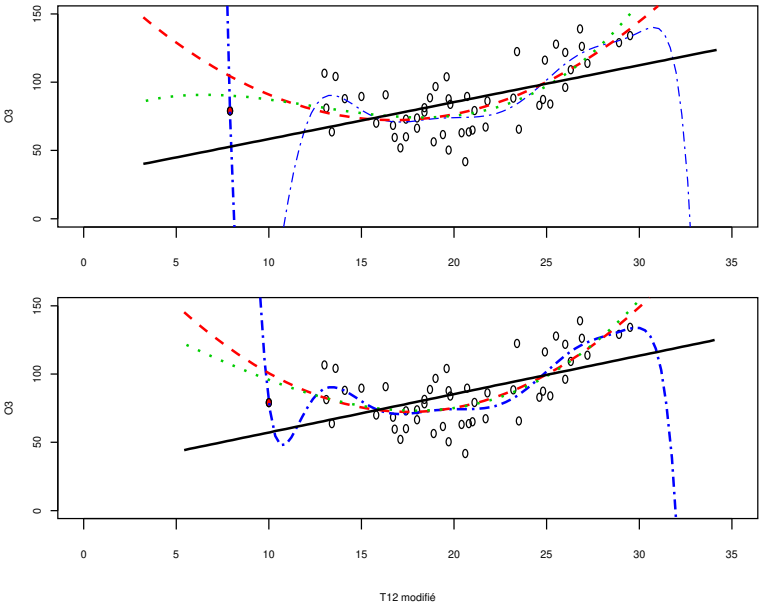


Fig. 16.3 – Sensibilité des estimateurs lors de la modification d'un point.

Modifions le point qui correspond au minimum de la température (7.9 °C) en le remplaçant par 10 °C et comparons les différents estimateurs. On peut observer

en figure 16.3 que les régressions linéaire, quadratique, cubique ne sont quasiment pas affectées, alors que celle avec un modèle polynomial de degré 9 se trouve fortement modifiée. Ainsi, lorsque le degré du polynôme est élevé, les estimations des coefficients ne sont pas robustes. Enlever une ou plusieurs données à l'échantillon peut modifier fortement les valeurs des estimations.

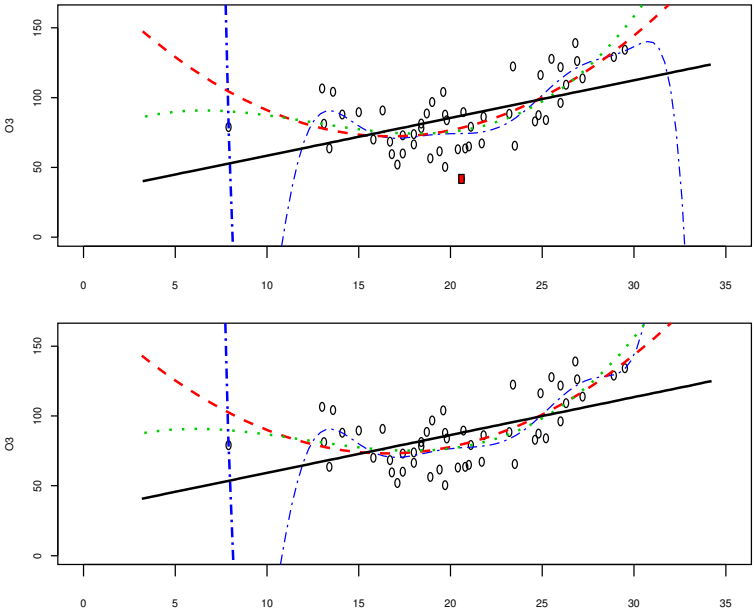


Fig. 16.4 – Evolution des estimateurs lors de la suppression d'un point.

Il faut retenir que la régression polynomiale est très facile à mettre en œuvre mais la solution est très dépendante du choix du degré d. De plus, l'estimateur des moindres carrés pour d grand est instable : des changements modiques dans les données peuvent entraîner des changements substantiels de la fonction de régression estimée.

En conclusion, sauf à soupçonner effectivement un modèle sous-jacent polynomial à degré élevé (ce qui est assez rare), nous préconisons d'utiliser la régression polynomiale avec un degré faible (inférieur ou égal à 3 par exemple). Si la solution obtenue ne convient pas car l'estimateur est trop rigide et ne semble pas adapté aux données, une autre solution est la régression par morceaux.

Plusieurs stratégies sont en fait possibles, et en particulier les deux suivantes :

1. découper l'étendue de la variable explicative en quelques morceaux (contigus). A l'intérieur de chaque intervalle, effectuer une régression polynomiale avec un degré faible : cela mènera à la régression spline décrite à la section 16.2 ;

2. considérer un grand nombre de points x situés dans l'étendue de la variable explicative. Pour chacun d'eux, déterminer une prédiction en effectuant une régression (constante, linéaire ou polynomiale) « locale » en pondérant les x_i de

sorte que les valeurs proches de x ont des poids plus élevés que ceux qui sont éloignés de x. Schématiquement, cela revient à faire des régressions sur des intervalles « glissants ». Cette technique correspond à la régression non paramétrique à noyaux décrite à la section 17.2.1.

16.2 Régression spline

16.2.1 Introduction

Dans le cas de la modélisation de la concentration de l'ozone par la température, l'étendue de la température est découpée en deux intervalles. Les commandes suivantes permettent d'effectuer deux régressions linéaires, une pour les données qui correspondent à une température inférieure à 23 °C, une pour les autres

```
> ind <- which(ozone[,2]<23)
> regd <- lm(O3~T12,data=ozone[ind,])
> regf <- lm(O3~T12,data=ozone[-ind,])
> gxd <- seq(3,23,length=50)
> gyd <- regd$coef[1]+gxd*regd$coef[2]
> gxf <- seq(23,35,length=50)
> gyf <- regf$coef[1]+gxf*regf$coef[2]
> plot(O3~T12,data=ozone)
> lines(gxd,gyd,col=2,lty=1,lwd=2)
> lines(gxf,gyf,col=2,lty=1,lwd=2)
> abline(v=23)
```

Nous obtenons alors le graphique suivant :

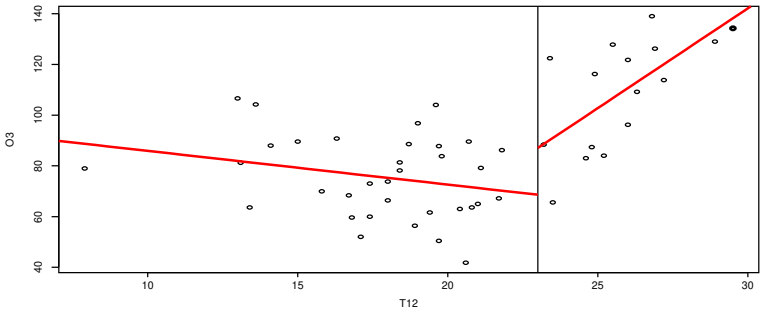


Fig. 16.5 – Régression par morceaux

La régression par morceaux est peu sensible à une modification d'un point de l'échantillon, et cela pour deux raisons :

1. S'il y a une modification, elle ne sera effective que sur l'intervalle auquel le point appartient.

2. En général, lorsque l'on effectue une régression par morceaux, on utilise des polynômes de degré faible qui sont plus robustes que des polynômes de degré élevé.

Alors que l'on peut penser que la fonction inconnue à estimer est continue, effectuer des régressions par morceaux donne en général des estimateurs discontinus aux jonctions des intervalles. La section suivante va proposer des contraintes afin d'obtenir des estimateurs continus voire dérivables.

16.2.2 Spline de régression

Reprenons le modèle défini dans la section précédente :

$$y_i = f(x_i) + \varepsilon_i.$$

La fonction f est toujours supposée inconnue mais nous formulons maintenant l'hypothèse qu'elle admet $d+1$ dérivées continues sur l'intervalle $[a,b]$ ($f \in \mathcal{C}^{d+1}_{[a,b]}$). Nous pouvons alors effectuer en tout point x de $[a,b]$ un développement limité avec reste intégral :

$$\begin{aligned} f(x) &= f(a) + \sum_{k=1}^{d} \frac{(x-a)^k}{k!} f^{(k)}(a) + \frac{1}{d!}\int_a^x (x-t)^d f^{(d+1)}(t)dt \\ &= \sum_{k=0}^{d} \alpha_k (x-a)^k + \frac{1}{d!}\int_a^x (x-t)^d f^{(d+1)}(t)dt \\ &= p(x) + r(x). \end{aligned}$$

Si le reste intégral $r(x)$ est petit, l'approximation de la fonction f par un polynôme sera correcte. On aura alors

$$f(x) \approx p(x) = \sum_{k=0}^{d} \alpha_k (x-a)^k = \sum_{k=0}^{d} \beta_k x^k.$$

Cependant, si on veut prendre d petit, il est fort possible que $r(x)$ ne soit pas négligeable. Dans ce cas, il faudra pouvoir estimer le reste $r(x)$. Pour cela, considérons la fonction $u_+ = \max(u,0)$ et réécrivons le reste intégral

$$\begin{aligned} r(x) &= \frac{1}{d!}\int_a^x (x-t)^d f^{(d+1)}(t)dt \\ &= \frac{1}{d!}\int_a^b (x-t)_+^d f^{(d+1)}(t)dt, \end{aligned}$$

où $(x-t)_+^d = [(x-t)_+]^d$ par définition. Il est impossible de calculer cette intégrale puisque f et ses dérivées sont inconnues. L'idée dès lors consiste à approximer cette intégrale par une somme de type Riemann. Pour ce faire, on découpe l'intervalle $[a,b]$ en définissant K points intérieurs à l'intervalle

$$a < \xi_1 < \xi_2 < \ldots < \xi_{K-1} < \xi_K < b.$$

Une estimation de l'intégrale est donnée par

$$\frac{1}{d!}\int_a^b (x-t)_+^d f^{(d+1)}(t)dt \approx \sum_{j=1}^{K} \beta_{d+j}(x-\xi_j)_+^d.$$

Nous avons alors l'approximation de f par

$$f(x) \quad \approx \quad s(x) = \sum_{k=0}^{d} \beta_k x^k + \sum_{j=1}^{K} \beta_{d+j}(x-\xi_j)_+^d.$$

La fonction s est appelée spline de degré d admettant K nœuds intérieurs. C'est par construction une fonction de $\mathcal{C}_{[a,b]}^{d+1}$. Définissons $\mathcal{S}_\xi^{d+1}$, l'espace vectoriel engendré par les fonctions $1, x, \ldots, x^d, (x-\xi_1)_+^d, \ldots, (x-\xi_K)_+^d$. Ces fonctions forment la base dite « base des puissances tronquées » (voir fig. 16.6 pour un exemple). $\mathcal{S}_\xi^{d+1}$, qui est de dimension $K+d+1$, contient l'ensemble des fonctions splines de degré d, admettant K nœuds intérieurs positionnés aux points $\xi_1, \cdots, \xi_K$.

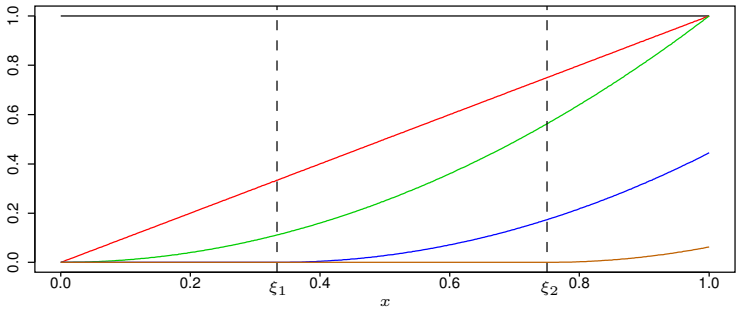


Fig. 16.6 – Base des puissances tronquées de S_ξ^3 sur $[0,1]$ avec $\xi = (1/3, 3/4)$.

Utiliser les splines de régression revient à substituer au modèle initial

$$y_i \quad = \quad f(x_i) + \varepsilon_i,$$

où $f \in \mathcal{C}_{[a,b]}^{d+1}$, le modèle simplifié

$$y_i \quad = \quad s(x_i) + \varepsilon_i,$$

où $s \in \mathcal{S}_\xi^{d+1}$. On peut aussi considérer qu'il s'agit de réécrire le modèle initial sous la forme

$$y_i \quad = \quad (f(x_i) - s(x_i)) + s(x_i) + \varepsilon_i$$

et supposer que l'erreur dite d'approximation $f-s$ est faible de sorte qu'en estimant la fonction s

$$s(x) \quad = \quad \sum_{k=0}^{d} \beta_k x^k + \sum_{j=1}^{K} \beta_{d+j}(x-\xi_j)_+^d \tag{16.1}$$

on approche la vraie fonction de régression f de façon satisfaisante.
Dès lors, lorsque le nombre de nœuds et leurs positions respectives sont fixés, on estime les paramètres β de l'équation (16.1) (appelée représentation de s dans la base des puissances tronquées) comme en régression linéaire multiple, en projetant orthogonalement Y sur $\mathcal{S}_\xi^{d+1}$. En notant X_{pt} la matrice de taille $n \times (d+K+1)$ composée des $d+K+1$ vecteurs de base ($X_{pt} = [1|x|\ldots|(x-\xi_K)_+^d]$), on obtient

$$\hat{\beta} \quad = \quad (X'_{pt}X_{pt})^{-1}X'_{pt}Y.$$

L'intérêt de la base des puissances tronquées est surtout pédagogique. Elle est peu utilisée dans la pratique car elle pose des problèmes de stabilité numérique (phénomène de Runge). Une autre base de $\mathcal{S}_\xi^{d+1}$ est en général programmée dans les logiciels de statistique : la base des différences divisées ou B-splines. Nous noterons les nouvelles fonctions de base $b_1(x), \ldots, b_{K+d+1}(x)$. L'avantage majeur de ces fonctions de base est leur caractère local, caractère que l'on peut voir sur le graphique suivant :

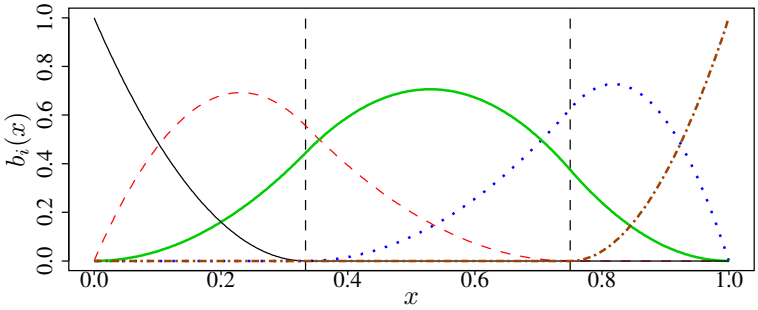


Fig. 16.7 – Graphique sur $[0,1]$ de la base des B-splines de S_ξ^3 avec $\xi = (1/3, 3/4)$.

Il est clair que les fonctions de base $b_j(t)$ sont non nulles sur l'intervalle $[\xi_j, \xi_{j+d+1}]$ et nulles en dehors, cela entraîne donc que $b_j(t)$ et $b_{j+d+1}(t)$ sont orthogonales. On peut montrer que $X'_B X_B$ est une matrice bande avec $X_B = [b_1(x)|\ldots|b_{K+d+1}(x)]$, (voir exercice 16.4). Cependant, pour pouvoir toujours écrire que les $b_j(t)$ sont non nulles sur l'intervalle $[\xi_j, \xi_{j+d+1}]$, il est d'usage d'étendre les nœuds intérieurs en un vecteur de nœuds étendus. Le vecteur des nœuds étendus est simplement constitué par le positionnement de $d+1$ nœuds fictifs aux bords de l'intervalle. Nous avons donc $d+1$ nœuds en a, puis les K nœuds intérieurs, puis $d+1$ nœuds en b. Graphiquement, nous pouvons représenter le vecteur des nœuds étendus de la façon suivante (voir fig. 16.8).

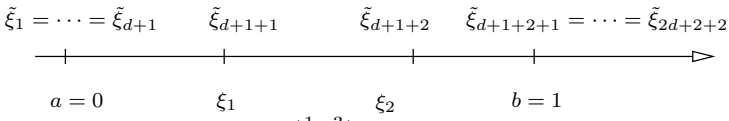


Fig. 16.8 – Nœuds intérieurs $\xi = (\frac{1}{3}, \frac{3}{4})$ et nœuds étendus pour l'intervalle $[0,1]$.

Avec l'aide de ces nœuds étendus, on peut écrire :

$$b_j(t) \quad \neq \quad 0 \quad \forall t \in [\tilde{\xi}_j, \tilde{\xi}_{j+d+1}].$$

Dans cette nouvelle base, la formule (16.1) devient :

$$s(x) \quad = \quad \sum_{k=1}^{d+1+K} \alpha_k b_k(x). \tag{16.2}$$

Dans l'exemple graphique (fig. 16.7), une base recouvre donc $d+1=3$ « intervalles » successifs, où les intervalles sont la partition de $[0,1]$ définie par les nœuds étendus. Cela illustre bien le caractère local de la base des B-splines.
La base des B-splines engendre bien évidemment le polynôme de degré 0 (ou constante « intercept ») mais il ne fait pas partie de la base (voir par exemple la figure 16.7). En statistique, il est d'usage d'incorporer le coefficient constant dans un modèle de régression. Dans ce cas, pour que l'hypothèse $\mathcal{H}_1$ (la matrice du plan d'expérience est de plein rang) soit valable, on ne peut pas ajouter à la matrice X_B le vecteur colonne $\mathbb{1}$. Il faudra donc soit supprimer une colonne de X_B, par exemple la première et la remplacer par le vecteur de $\mathbb{1}$ (`intercept=FALSE`, argument par défaut de la fonction `bs` de R), soit garder toutes les fonctions de base (argument `intercept=TRUE`).
Pour effectuer une régression spline, il faut donc choisir une suite de nœuds intérieurs, un degré et les bords de l'intervalle (par défaut le logiciel prend le minimum et le maximum de l'échantillon). Pour l'exemple de l'ozone, choisissons deux nœuds intérieurs (15 et 23 par exemple), des polynômes de degré 2 et prenons 5 et 32 comme bords de l'intervalle. Il faut d'abord transformer la variable explicative dans sa base des B-splines grâce aux commandes suivantes :

```
> library(splines)
> XB <- bs(ozone[,2], knots=c(15,23), degree=2,
                                  Boundary.knots=c(5,32))
```

La matrice `XB` a $K+d$ colonnes, la première étant supprimée (`intercept=FALSE` par défaut). Il suffit ensuite d'effectuer la régression de la variable à expliquer comme suit :

```
> regs <- lm(ozone[,"O3"] ~ XB)
> regs$coef
(Intercept)         XB1         XB2         XB3         XB4
  51.101947   61.543761    5.562286   70.459103  106.711539
```

Nous avons donc ici une régression spline avec 5 paramètres. Pour avoir une représentation graphique de la fonction estimée, on choisit une grille de points où la fonction va être évaluée. Il faut évaluer toutes les fonctions de la base des B-splines utilisée pour la régression en chaque point de cette grille. Il faut ensuite appliquer la formule (16.2) et dessiner la fonction estimée. Le coefficient constant ne faisant pas partie de la base, il est traité à part.

```
> grillex <- seq(5,32,length=100)
> bgrillex <- bs(grillex, knots=c(15,23), degree=2,
                                  Boundary.knots=c(5,32))
> prev <- bgrillex%*%as.matrix(regs$coeff[-1])+regs$coeff[1]
> plot(O3~T12,data=ozone)
> lines(grillex,prev,col=2)
> abline(v=c(15,23))
```

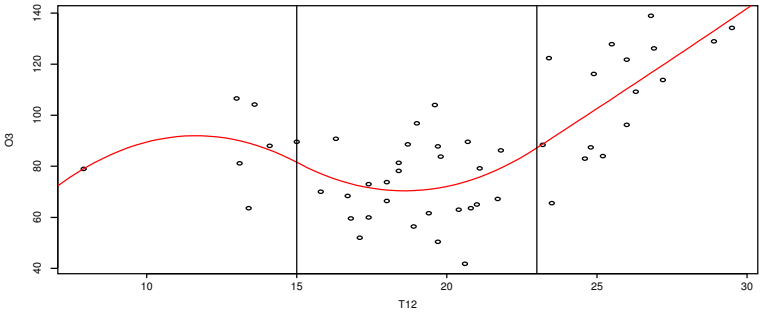


Fig. 16.9 – 50 données journalières de température et d'ozone avec l'estimation d'une régression spline avec 2 nœuds intérieurs et de degré 2 soit 5 paramètres.

La figure 16.9 nous confirme que nous avons effectué une régression polynomiale par morceaux avec des contraintes de raccordement aux nœuds.

16.3 Spline de lissage

Spline est un mot anglais qui désigne une latte en bois flexible qui était utilisée par les dessinateurs pour tracer des courbes passant par des points fixés.
Pour passer par ces points, la latte de bois se déformait et le tracé réalisé minimisait donc l'énergie de déformation de la latte (fig. 16.10). Par analogie, ce terme désigne des familles de fonctions d'interpolation ou de lissage présentant des propriétés optimales de régularité.

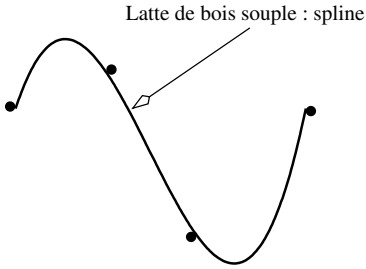


Fig. 16.10 – Spline en bois.

Dans un cadre de régression, une spline de lissage est une fonction f qui possède à la fois une bonne qualité d'ajustement ($f(x_i)$ proche de y_i) et de bonnes propriétés de régularité (par exemple une dérivée seconde qui ne prend pas des valeurs trop élevées). Le problème mathématique va donc consister à trouver la fonction f qui minimise

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int \left(f^{(2)}(t)\right)^2 dt \tag{16.3}$$

où $\lambda \geq 0$ est un paramètre à calibrer. La première partie mesure la proximité avec les données tandis que la seconde pénalise la forme (plus la fonction f sera oscillante et plus l'intégrale sera élevée). Ce problème est proche des problèmes de minimisation ridge et lasso étudiés dans le chapitre 8. Nous ne rentrons par dans les détails techniques de résolution de ce problème de minimisation mais il est possible de montrer que, pour une valeur de $\lambda \geq 0$ fixée, il existe une solution unique que l'on peut calculer.
Comme pour les régressions ridge et lasso, le choix de λ est crucial. En effet un choix de λ « grand » va impliquer que le terme de pénalisation va dominer et pour minimiser le critère il faudra trouver un estimateur tel que le terme dans l'intégrale soit petit, c'est-à-dire une fonction qui oscille peu. La fonction R qui permet d'effectuer cette régression est **smooth.spline**. Pour une forte valeur de λ dans l'argument `lambda`

```
> regssplinel1 <- smooth.spline(ozone[,2],ozone[,1],lambda =100)
> prevl1 <- predict(regssplinel1,grillex)
> plot(O3~T12,data=ozone)
> lines(prevl1$x,prevl1$y,col=2)
```

nous obtenons la figure 16.11.

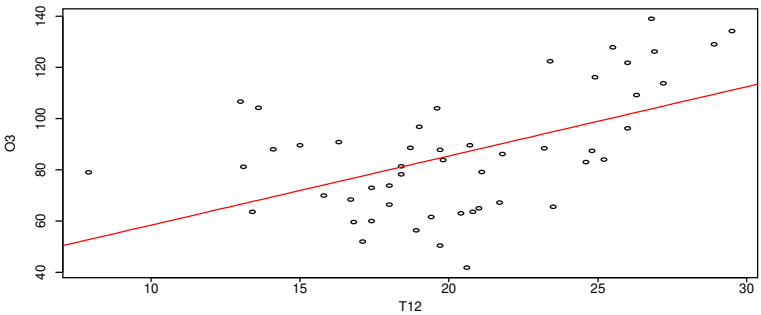


Fig. 16.11 – 50 données journalières de température et d'ozone avec l'estimation de la fonction de régression par des splines de lissage pour une forte valeur de λ.

Réciproquement, une valeur de λ (trop) petite va favoriser le terme associé à la proximité aux données, c'est-à-dire le premier terme de (16.3). Nous risquons alors

d'obtenir une courbe très variable qui va se rapprocher des points de l'échantillon (risque de sur-ajustement). Vérifions cela avec le même code que précédemment mais avec $\lambda = 10^{-6}$ (voir fig. 16.12).

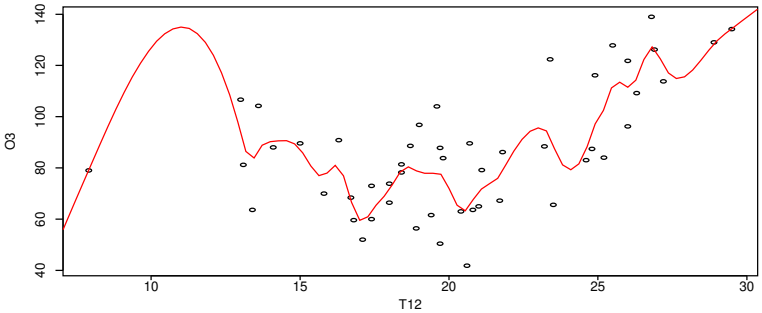


Fig. 16.12 – 50 données journalières de température et d'ozone avec l'estimation de la fonction de régression par des splines de lissage pour une faible valeur de λ.

Le choix du paramètre λ se fait généralement par validation croisée. La fonction `smooth.spline` intégrée dans R est utilisable de la manière suivante :

```
> regsspline <- smooth.spline(ozone[,2],ozone[,1])
> prev <- predict(regsspline,grillex)
> plot(O3~T12,data=ozone)
> lines(prev$x,prev$y,col=2)
```

Le code ci-dessus donne la figure 16.13.

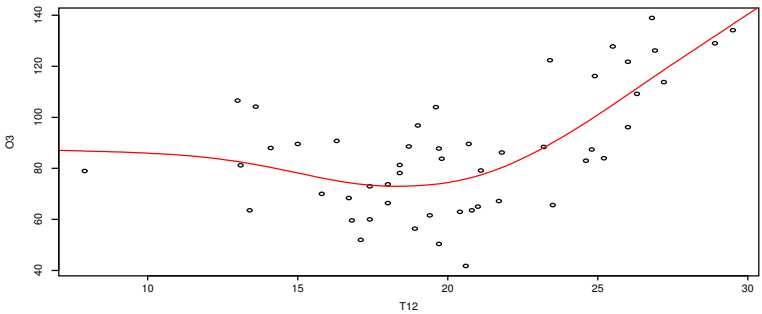


Fig. 16.13 – 50 données journalières de température et d'ozone avec l'estimation de la fonction de régression par des splines de lissage.

Dans le résumé de la fonction nous avons :

```
> regsspline
Call:
smooth.spline(x = ozone[, 2], y = ozone[, 1])

Smoothing Parameter spar=0.941034 lambda=0.0068333 (15 iterations)
Equivalent Degrees of Freedom (Df): 4.156771
Penalized Criterion: 11036.88
GCV: 289.8012
```

et nous voyons le `lambda` optimal proposé par la fonction ainsi qu'un terme donnant le nombre de degré de liberté équivalent. Ce terme permet de mesurer une forme de complexité de l'estimateur obtenu. Dans le modèle de régression linéaire multiple, la complexité est généralement définie par le nombre de variables dans le modèle. Ce nombre correspond à la dimension de l'espace sur lequel on projette, et donc à la trace du projecteur P_X. On rappelle que ce projecteur vérifie $\hat{Y} = P_X Y$. Dans le cas des splines de lissage, il n'existe pas de matrice de projection mais il existe une matrice S_λ, appelée matrice de lissage, qui vérifie

$$\hat{Y} = S_\lambda Y.$$

Par analogie avec le modèle linéaire, le nombre de paramètres ou le nombre de degrés de liberté équivalent d'une spline de lissage est la trace de cette matrice S_λ. Ici, le nombre de degrés équivalent (Df) vaut 4.16.

16.4 Exercices

Exercice 16.1 (Questions de cours)

1) La difficulté pour effectuer une régression polynomiale réside dans
   A. le choix du degré,
   B. le choix des données,
   C. l'interprétation du modèle.

2) La régression spline est une régression polynomiale par morceaux
   A. avec contraintes aux nœuds,
   B. sans contrainte aux nœuds,
   C. ce n'est pas une régression.

3) Vous effectuez une régression spline de degré 3 avec 2 nœuds intérieurs (différents) et une régression spline de degré 1 avec 4 nœuds intérieurs (tous différents les uns des autres). Avez-vous le même nombre de paramètres ?
   A. Oui.
   B. Non.
   C. Cela dépend de l'emplacement des nœuds.

4) Vous effectuez une régression spline de degré 3 avec 2 nœuds intérieurs et une régression spline de degré 1 avec 4 nœuds intérieurs. Avez-vous les mêmes résultats ?
   A. Oui.
   B. Non.
   C. Cela dépend de l'emplacement des nœuds.

Exercice 16.2 (fonction polyreg)

1) Calculez l'écart type empirique de la variable `T12` du tableau `ozone` (fonction `sd`).
2) Créez un vecteur nommé `grillex` de 100 points régulièrement répartis entre le minimum de `T12` moins un écart type et le maximum de `T12` plus un écart type (fonction **seq**).
3) Transformez ce vecteur en data-frame (fonction **data.frame**) et affectez comme nom de colonne (`names`) le nom `T12`.
4) Effectuez une régression polynomiale de degré 3 grâce aux fonctions `lm` et `poly`. Les polynômes seront choisis non orthogonaux (argument `raw=TRUE`). L'aide de `cars` pourra aussi être consultée.
5) Prévoyez sur la grille grâce au data-frame de la question 3) et au modèle de la question précédente (`predict`).
6) Proposez une fonction (nommée **polyreg**) qui possède comme arguments un data-frame de données, le degré d qui par défaut sera 3 et qui retourne une liste de deux arguments : le vecteur `grillex` et le vecteur de prévision. Les noms de variables ne seront pas des arguments.

Exercice 16.3

En utilisant les données `ozone.txt` et la fonction **polyreg**, retrouvez les commandes permettant d'obtenir le graphique 16.2.

Exercice 16.4

Considérons la matrice X_B du plan d'expérience obtenue à partir d'une variable réelle X transformée dans $\mathcal{S}_\xi^{d+1}$. Cette matrice de taille $n \times (d+K+1)$ est composée des $d+K+1$ vecteurs de base. Démontrer que $X_B' X_B$ est une matrice bande. Que peut-on dire sur la corrélation des $\hat{\beta}_k$ estimateur des MC dans cette base ?

Chapitre 17

Estimateurs à noyau et k plus proches voisins

Nous présentons dans ce chapitre une courte introduction sur les estimateurs à noyau et les estimateurs des k plus proches voisins. Ils sont souvent vus comme les estimateurs non paramétriques de référence. Le lecteur souhaitant approfondir ses connaissances pourra consulter le livre de Eubank (1999).

17.1 Introduction

Reprenons l'exemple de l'ozone : nous allons chercher à expliquer la concentration en ozone (`O3`) par la température à 12 h (`T12`).

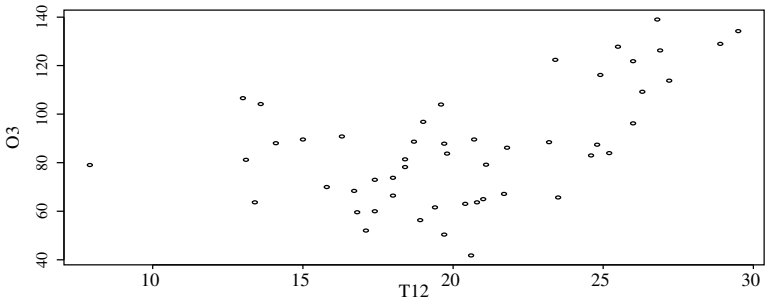


Fig. 17.1 – 50 données journalières de température et `O3`.

Chaque point du graphique représente, pour un jour donné, une mesure de la température à 12 h et le pic d'ozone de la journée. En examinant la figure 17.1 nous voyons que les points ne sont pas répartis autour d'une droite et qu'il semble y avoir une cassure après 20 °C : un ajustement linéaire n'est donc pas adéquat.

Nous proposons donc de considérer le modèle de régression

$$y_i = f(x_i) + \varepsilon_i \quad i = 1, \cdots, n,$$

où y_i représente la i^{e} réponse et donc ici la concentration en ozone et ε_i est une variable aléatoire telle que $\mathbb{E}(\varepsilon_i) = 0$. Par conséquent, l'espérance $\mathbb{E}(y_i)$ est vue comme une fonction (inconnue) $f(x_i)$ de la i^{e} température à 12 h x_i. L'objectif de ce chapitre est d'autoriser f à appartenir à une classe plus large et plus flexible de fonctions que les fonctions affines ou polynomiales.
Sans perte de généralités, on suppose que les x_i sont ordonnés : $x_1 \leq x_2 \leq \ldots \leq x_n$. Nous souhaitons dans un premier temps proposer que la fonction inconnue soit une fonction en escalier, le problème de minimisation s'écrit :

$$\underset{f \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - f(x_i)\right)^2, \tag{17.1}$$

où $\mathcal{G}$ est maintenant l'espace des fonctions en escalier. Pour définir un peu mieux cet ensemble $\mathcal{G}$, nous avons (en gros) deux possibilités. La première consiste à définir à priori et sans qu'il ne soit plus possible de revenir dessus, les $m-1$ emplacements des marches d'escalier (ou la partition de l'intervalle $[x_1, x_n]$) : $\xi_1 = x_1 < \xi_2 < \xi_3 < \ldots < \xi_m < x_n = \xi_{m+1}$. Une fonction en escalier consistera donc à choisir les hauteurs de marche en ces points ξ_j. Plus formellement, l'ensemble $\mathcal{G}$ sera donc l'ensemble des fonctions

$$\left\{ f : [x_1, x_n] \mapsto \mathbb{R} \mid f(x) = \sum_{j=1}^{m} c_j \mathbf{1}_{I_j}(x), \quad c_j \in \mathbb{R}, j \in \{1, \ldots, m\} \right\}$$

où I_j désigne l'intervalle $]\xi_j, \xi_{j+1}]$. Dans ce cadre, la minimisation revient donc à chercher les hauteurs $\hat{c}_j$ qui minimise le critère (17.1).
Une seconde possibilité pour définir $\mathcal{G}$ serait de laisser à la fois les $\{\xi_j\}_{j=2}^m$ et les $\{c_j\}_{j=1}^m$ libres (et donc à optimiser). Cette approche plus complexe ne sera pas évoquée.
Puisque nous nous concentrons sur la première possibilité, il nous faut définir les « emplacements des marches » de l'escalier $\{\xi_j\}_{j=2}^m$. Comme nous avons 50 points, nous proposons de considérer $m = 5$ segments comprenant 10 points chacun et nous approchons donc f par une valeur constante sur chaque segment avec les segments suivants :

$$I_1 = [x_1, x_{10}], \quad I_2 =]x_{10}, x_{20}], \quad \ldots, \quad I_5 =]x_{40}, x_{50}].$$

Concernant notre exemple, nous ordonnons les données en utilisant R grâce à :

```
> ind <- order(ozone[,"T12"])
> T12o <- ozone[ind,"T12"]
> O3o <- ozone[ind,"O3"]
```

Une fonction de la classe $\mathcal{G}$, constante sur chaque intervalle, s'écrit

$$f(x) = \sum_{j=1}^{5} c_j \mathbf{1}_{I_j}(x).$$

Le problème de minimisation (17.1) devient

$$\underset{c_1,\ldots,c_5}{\operatorname{argmin}} \sum_{j=1}^{5} \sum_{i=1}^{n} \mathbf{1}_{I_j}(x_i)(y_i - c_j)^2.$$

Il peut également s'écrire comme une problème de régression pondérée en introduisant les poids :

$$\underset{c_1,\ldots,c_5}{\operatorname{argmin}} \sum_{j=1}^{5} \sum_{i=1}^{n} w_{ij}(y_i - c_j)^2, \quad \text{avec } w_{ij} = \begin{cases} 1 \text{ si } x_i \in I_j, \\ 0 \text{ sinon.} \end{cases}$$

Chercher la meilleure approximation constante sur chaque segment I_j revient donc à effectuer une régression constante sur les segments. L'estimateur d'une régression constante étant la moyenne, on déduit que les $\hat{c}_j$ s'obtiennent en faisant la moyenne des y_i pour lesquels les x_i sont dans l'intervalle I_j. Nous pouvons ainsi calculer les $\hat{c}_j$ avec les commandes suivantes :

```
> reg1 <- lm(O3o~1,weight=c(rep(1,10),rep(0,40)))
> reg2 <- lm(O3o~1,weight=c(rep(0,10),rep(1,10),rep(0,30)))
> reg3 <- lm(O3o~1,weight=c(rep(0,20),rep(1,10),rep(0,20)))
> reg4 <- lm(O3o~1,weight=c(rep(0,30),rep(1,10),rep(0,10)))
> reg5 <- lm(O3o~1,weight=c(rep(0,40),rep(1,10)))
```

et nous obtenons alors un estimateur en escaliers de la fonction f (voir fig. 17.2).

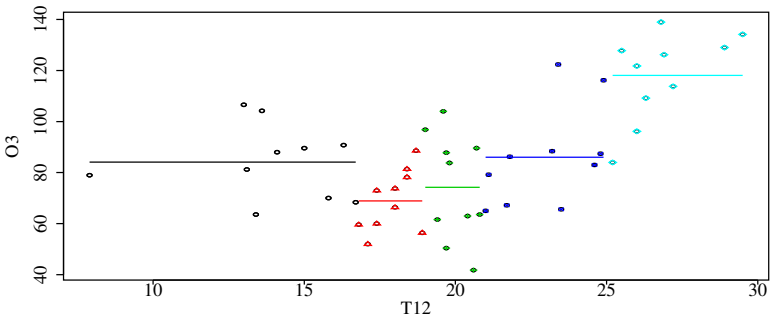


Fig. 17.2 – Estimation par morceaux.

Un tel estimateur est semblable à un estimateur de type histogramme pour le problème d'estimation de la densité. Tout comme l'histogramme, il possède l'inconvénient de ne pas être continu, il y a un saut lorsqu'on passe d'un intervalle

à un autre. Les sauts dépendent ainsi du choix des segments qui est effectué par l'utilisateur. Cela peut se révéler gênant dans la mesure où aucune considération métier ne nous pousse à penser que la concentration en ozone doit être discontinue entre la 10e et la 11e température, puis entre la 20e et la 21e, etc.
De plus, cette façon de procéder pose un problème évident lors de la prédiction. Ainsi pour prédire Y au niveau du dixième point par exemple, nous utilisons les points $x_1, \ldots, x_{10}$ dont certains comme x_1 en sont très éloignés alors qu'il aurait semblé plus judicieux de considérer les points x_{11} ou x_{12} beaucoup plus proches de lui. Il paraît donc naturel de proposer une méthode qui choisisse de façon « intelligente » le poids des observations entrant dans la prévision de Y pour une valeur de x quelconque. L'estimateur à noyau, qui est basé sur une idée proche de cet estimateur en escalier, permet de pallier ce problème de discontinuité.

17.2 Estimateurs par moyennes locales

De nombreux estimateurs s'écrivent comme une moyenne pondérée des y_i

$$\hat{f}(x) = \sum_{i=1}^{n} W_{ni}(x) y_i$$

où $W_{ni}(x)$ représente le poids accordé à la ie valeur de Y. Les moyennes locales consistent à définir les $W_{ni}(x)$ en fonction de la proximité entre l'observation x_i et le point x où on cherche à estimer la fonction de régression. L'idée sous-jacente est de donner un poids plus important aux observations qui sont proches de x. Nous présentons dans cette section deux types de poids basés sur cette idée.

17.2.1 Estimateurs à noyau

Pour améliorer notre estimateur en escalier, nous proposons ici une façon plus astucieuse de choisir les segments en considérant que tous les points x où on cherche à estimer la fonction de régression f doivent être au centre du segment. Dans ce cas, il n'y a plus besoin de fixer des segments mais seulement une longueur de segment. Si on considère par exemple une longueur égale à $2h$ avec $h > 0$, nous allons estimer f en x en considérant uniquement les observations x_i qui appartiennent au segment $[x-h, x+h]$. Ainsi un estimateur naturel pour $f(x)$ est alors la moyenne des réponses y_i pour lesquelles les x_i appartiennent à l'intervalle $[x-h, x+h]$

$$\hat{f}_h(x) = \frac{\sum_{i=1}^{n} \mathbf{1}_{[x-h,x+h]}(x_i) y_i}{\sum_{i=1}^{n} \mathbf{1}_{[x-h,x+h]}(x_i)},$$

à condition que le dénominateur soit strictement supérieur à 0. Il est intéressant de noter que cet estimateur est la solution d'un problème de type moindre carrés pondérés. En effet, pour une valeur de x fixée, la constante β qui minimise

$$\sum_{i=1}^{n} (y_i - \beta)^2 \mathbf{1}_{[x_i-h,x_i+h]}(x) \tag{17.2}$$

est $\hat{f}_h(x)$. Cette valeur s'obtient donc en effectuant une régression constante sur le segment $[x-h, x+h]$ et il est facile de voir que la solution est donnée par la moyenne des y_i tels que $x_i \in [x-h, x+h]$. La figure 17.3 illustre ce procédé d'estimation pour $x = 16$ et $h = 2$. On considère donc uniquement les observations dans l'intervalle $[14, 18]$ et l'estimateur de f en $x = 16$ correspond à la concentration en ozone moyenne des observations qui appartiennent à cet intervalle. Cette moyenne vaut 71.96.

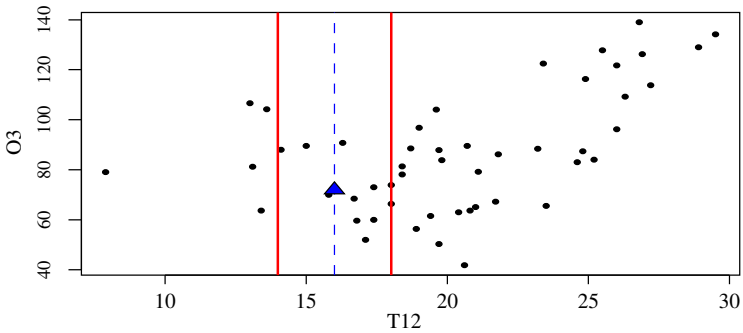


Fig. 17.3 – Points utilisés pour calculer l'estimateur au point 16. La valeur de l'estimateur en ce point est représentée par le triangle.

La solution du problème (17.2) est la moyenne des observations dans le segment $[x-h, x+h]$ et s'écrit

$$\hat{f}_h(x) = \frac{1}{|\{i : x_i \in [x-h, x+h]\}|} \sum_{i:x_i \in [x-h,x+h]} y_i = \frac{\sum_{i=1}^n y_i \mathbf{1}_{[x-h,x+h]}(x_i)}{\sum_{i=1}^n \mathbf{1}_{[x-h,x+h]}(x_i)}. \quad (17.3)$$

Remarquons que dans le problème de minimisation (17.2), toutes les observations qui se trouvent dans la fenêtre $[x-h, x+h]$ ont le même poids. Il peut être intéressant de donner des poids différents aux observations en fonction de leur proximité avec x par exemple. Cela s'effectue en remplaçant l'indicatrice par une fonction K, appelée noyau, qui est le plus souvent une densité de probabilité (fonction positive dont l'aire sous la courbe vaut 1). Le problème de minimisation devient

$$\underset{\beta \in \mathbb{R}}{\operatorname{argmin}} \sum_{i=1}^n (y_i - \beta)^2 K\left(\frac{x - x_i}{h}\right) \quad (17.4)$$

et la solution est donnée par

$$\hat{f}_h(x) = \frac{\sum_{i=1}^n y_i K\left(\frac{x-x_i}{h}\right)}{\sum_{i=1}^n K\left(\frac{x-x_i}{h}\right)}. \quad (17.5)$$

Cet estimateur, appelé estimateur à noyau ou de Nadaraya Watson, dépend de deux paramètres : un réel positif h appelé **fenêtre** et une fonction K appelée **noyau**.

— **La fenêtre** est un réel positif h. Elle permet à l'utilisateur de contrôler le caractère local des poids. En effet plus la fenêtre est petite et plus on va se focaliser sur les observations très proches de x pour calculer l'estimateur. Par exemple, pour l'estimateur (17.3), la fenêtre contrôle la longueur de l'intervalle centré en x et par conséquent le nombre d'observations qui sont prises en compte pour calculer l'estimateur en x.

— **Le noyau** est une fonction $K : \mathbb{R} \to \mathbb{R}^+$ positive dont l'aire sous la courbe vaut 1 (densité de probabilité). Cette fonction permet de contrôler le poids donné à chaque observation en fonction de la proximité entre x et x_i. Elle prendra donc généralement de fortes valeurs lorsque la distance entre x et x_i est petite et des faibles valeurs lorsque cette distance est grande. Le noyau proposé dans l'écriture (17.3) est $K(u) = \frac{1}{2}\mathbf{1}_{[-1,1]}(u)$. Ce noyau, appelé noyau uniforme, donne le même poids à toutes les observations dans la fenêtre. Par conséquent, des points situés dans la fenêtre, mais éloignés de x, auront des poids identiques à des points très proches de x.

D'autres noyaux peuvent être utilisés pour donner plus de poids aux observations proches de x. On citera par exemple le noyau gaussien

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right)$$

ou celui d'Epanechnikov

$$K(u) = \frac{3}{4}(1 - u^2)\mathbf{1}_{[-1,1]}(u).$$

Ces noyaux sont représentés sur la figure 17.4.

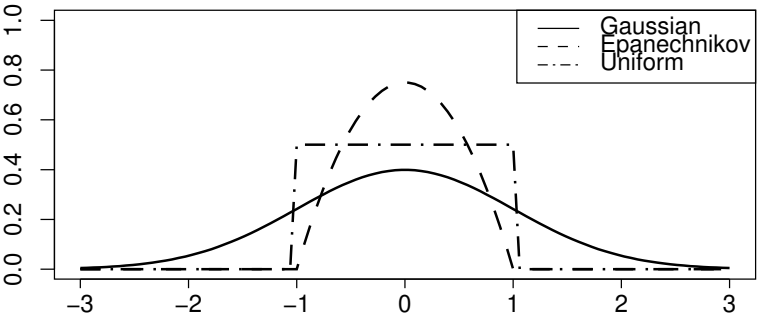


Fig. 17.4 – Représentation des noyaux gaussien, Epanechnikov et uniforme.

Une extension naturelle au cas multivarié $x \in \mathbb{R}^p$ consiste simplement à utiliser un noyau $K : \mathbb{R}^p \to \mathbb{R}^+$ (pour une définition plus précise, voir section 17.4.1, p. 406). La fonction **npregress** du package **ibr** permet de calculer des estimateurs à noyau. On représente les estimateurs à noyau pour 3 valeurs de fenêtres différentes avec le code suivant (voir figure 17.5) :

```
> library(ibr)
> x <- seq(7,30,by=0.01)
> par(mfrow=c(1,3))
> h <- c(20,3,0.05)
> for (i in h){
+   plot(T12,O3,pch=20,xlab="T12",ylab="O3")
+   tmp <- npregress(T12,O3,bandwidth = i)
+   prev <- predict(tmp,newdata=x)
+   lines(x,prev,col="blue",lwd=2)
+ }
```

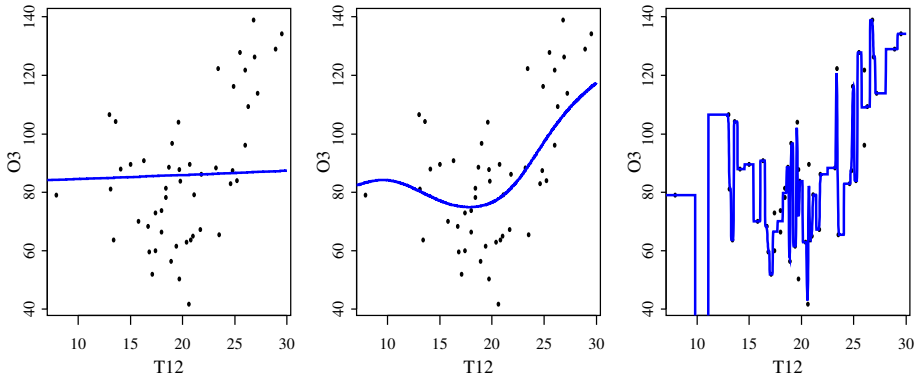


Fig. 17.5 – Estimateurs à noyau pour $h = 20$ (gauche), 3 (milieu) et 0.05 (droite).

Le noyau utilisé ici est le noyau gaussien, on peut le changer en modifiant l'option `kernel` dans **npregress**. On remarque que le choix de la fenêtre a une importance cruciale sur la performance de l'estimateur. Ce choix sera abordé dans la section 17.4.2.

Le caractère **local** de l'estimateur est contrôlé par la fenêtre et ce choix est basé sur une distance entre les points x_i et x. En général, la fenêtre h est constante pour l'estimation de la fonction, elle ne dépend pas de x. Si la répartition des points en fonction de la variable X n'est pas uniforme (comme dans le cas de la température), il y aura des endroits où la fonction va être estimée avec beaucoup de points (s'il y a beaucoup de points dans l'intervalle) et au contraire des endroits où la fonction va être estimée avec peu de points (par exemple dès que la température est en dessous de 15 degrés, ou plus généralement aux bords du domaine). Il est possible de décider de prendre toujours le même nombre k de points pour estimer la fonction et ceci en tout point du domaine. Le caractère local est dans ce cas lié à ce nombre de voisins k. Cet estimateur, appelé estimateur des k plus proches voisins, est présenté dans la section suivante.

17.2.2 Les k plus proches voisins

Il est facile de voir que l'estimateur à noyau (17.5) peut se mettre sous la forme

$$\hat{f}_h(x) = \sum_{i=1}^{n} W_{ni}(x) y_i \quad \text{avec} \quad W_{ni}(x) = \frac{K\left(\frac{x-x_i}{h}\right)}{\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)}.$$

Les poids sont ici définis à partir de la distance entre x et les x_i. Une autre façon de procéder pour définir les poids consiste à utiliser la position de x_i relativement aux autres observations : on donnera un poids important à la i^{e} observation uniquement si x_i fait partie des plus proches voisins de x.
Formellement, étant donné $(x_1, y_1), \ldots, (x_n, y_n)$ avec $x_i \in \mathbb{R}, y_i \in \mathbb{R}$ et $x \in \mathbb{R}$, on réordonne les observations en fonction de leur proximité avec x :

$$(x_{(1)}(x), y_{(1)}(x)), \ldots, (x_{(n)}(x), y_{(n)}(x))$$

avec $|x_{(1)}(x) - x| \leq \ldots \leq |x_{(n)}(x) - x|$. En cas d'égalité, une règle doit être fixée. Il en existe plusieurs, on peut par exemple utiliser les indices : si x_i et x_j sont à égale distance de x, i est déclaré plus proche si $i < j$. Etant donné k un entier inférieur ou égal à n, l'estimateur des k plus proches voisins s'obtient en faisant la moyenne des y_i qui figurent parmi les k plus proches voisins de x :

$$\hat{f}_k(x) = \frac{1}{k} \sum_{i=1}^{k} y_{(i)}(x).$$

On remarquera que, là encore, cet estimateur s'écrit comme une moyenne locale $\sum_{i=1}^{n} W_{ni}(x) y_i$. Les poids sont ici égaux à $1/k$ si x_i appartient aux k plus proches voisins de x, 0 sinon.
Une généralisation multivariée s'obtient assez simplement en remplaçant l'ordre $|x_{(1)}(x) - x| \leq \ldots \leq |x_{(n)}(x) - x|$ (qui n'est pas défini quand $x \in \mathbb{R}^p$) par l'ordre $\|x_{(1)}(x) - x\| \leq \ldots \leq \|x_{(n)}(x) - x\|$ (voir section 17.4.1 p. 406).
Sur R, on peut utiliser la fonction **knn.reg** du package **FNN** pour calculer des estimateurs par plus proches voisins. On propose par exemple ici de représenter ces estimateurs pour $k = 1, 10$ et 50 (voir figure 17.6) :

```
> par(mfrow=c(1,3))
> library(FNN)
> k <- c(50,10,1)
> for (i in k){
+   mod <- knn.reg(train=T12o,test=as.matrix(x),y=O3o,k=i)
+   plot(T12o,O3o,pch=20,xlab="T12",ylab="O3")
+   lines(x,mod$pred,col="blue",lwd=2)
+ }
```

Le choix du nombre de voisins k joue un rôle identique pour l'estimateur des k plus proches voisins au choix de la fenêtre h pour l'estimateur à noyau. Ainsi pour

de grandes valeurs de k ou de h, les estimateurs utiliseront une large étendue des données et seront donc stables (faible variance) alors que pour un choix de petites valeurs de k ou h, les estimateurs n'utiliseront qu'un petit nombre de données et seront donc très variables (forte variance).

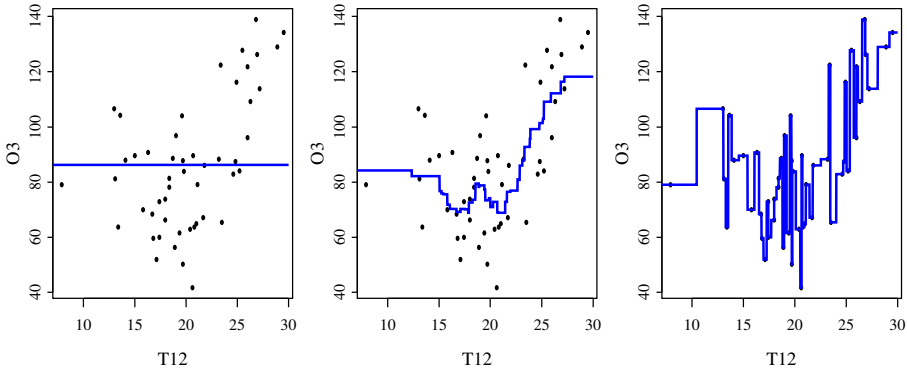


Fig. 17.6 – Estimateurs des k-ppv pour $k = 50$ (gauche), 10 (milieu), 1 (droite).

17.3 Choix des paramètres de lissage

Les figures 17.5 et 17.6 montrent que les paramètres de lissage (h pour le noyau et k pour les plus proches voisins) sont très importants pour les performances des estimateurs. Nous noterons donc λ ce paramètre de lissage qui sera soit h soit k selon l'estimateur considéré. Les approches classiques permettant de choisir ces paramètres sont proches des techniques présentées au chapitre 10. Elles consistent à se donner un critère de prévision, une grille de valeurs (pour h ou k) et à choisir la valeur de la grille qui optimise le critère calculé.

Dans le cas de ces deux estimateurs non paramétriques, il existe une astuce qui permet de faire de façon efficace de la validation croisée leave-one-out (voir algorithme 3 avec $K = n$).

Avant de présenter cette astuce, remarquons que, comme nous l'avons vu pour les splines (voir en fin de section 16.3, p. 390), les estimateur à noyau et des k plus proches voisins sont des lisseurs : il existe une matrice $S(X, \lambda)$ de taille $n \times n$, appelée matrice de lissage, telle que

$$\hat{Y} = S(X, \lambda)Y.$$

Pour l'estimateur à noyau, la matrice de lissage a pour terme général

$$S_{ij}(X, h) = \frac{K((x_i - x_j)/h)}{\sum_l K((x_i - x_l)/h)}. \tag{17.6}$$

Pour l'estimateur des k plus proches voisins, elle est donnée par

$$S_{ij}(X,k) \quad = \quad \begin{cases} 1/k & \text{si } x_j \text{ est parmi les } k\text{-ppv de } x_i \\ 0 & \text{sinon.} \end{cases}$$

Comme pour les splines de lissage (voir section 16.3, p. 390) cette présentation des estimateurs à noyau et des k plus proches voisins avec une matrice de lissage $S(X,\lambda)$ permet d'introduire le nombre de paramètres ou le nombre de degrés de liberté équivalent. Il est défini comme étant la trace de cette matrice de lissage $\mathrm{df} = \mathrm{tr}(S(X,\lambda))$. Rappelons que ce nombre de degrés de liberté équivalent provient d'une simple analogie avec la régression linéaire : en régression linéaire nous avons $\hat{Y} = P_X Y$, donc la matrice de lissage est dans ce cas P_X. La trace d'un projecteur est égale à la dimension du sous espace dans lequel on projette, qui, lorsque X est de plein rang, vaut p, le nombre de paramètres. On a donc qu'en régression linéaire, le nombre de paramètres vaut $\mathrm{tr}\, P_X$. Cette formulation se généralise et donne lieu au nombre de degrés de liberté équivalent.
Revenons au problème de sélection du paramètre du paramètre de lissage λ. Nous proposons d'utiliser la validation croisée leave-one-out. Cette approche consiste à trouver sur une grille de valeurs de λ celle qui minimise le critère suivant

$$\mathrm{LOO}(\hat{f}_\lambda) = \sum_{i=1}^{n} (y_i - \hat{f}_\lambda^i(x_i))^2,$$

$\hat{f}_\lambda^i$ désigne l'estimateur $\hat{f}_\lambda$ (noyau ou k plus proches voisins) calculé sans la i^{e} observation : la i^{e} observation est enlevée du jeu de données et l'estimation est alors conduite avec ce jeu de données de taille $n-1$. De prime abord, l'approche paraît coûteuse en temps de calcul car pour chaque λ de la grille, il faut faire n estimations avec $n-1$ points, prévoir puis calculer le critère et enfin choisir la valeur (optimale) qui minimise le critère parmi tous les λ de la grille.
Néanmoins pour les estimateurs à noyau et des plus proches voisins, on montre dans l'exercice 17.6 que

$$\mathrm{LOO}(\hat{f}_h) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{y_i - \hat{f}_h(x_i)}{1 - S_{ii}(X,h)} \right)^2 \tag{17.7}$$

et

$$\mathrm{LOO}(\hat{f}_k) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{y_i - \hat{f}_{k+1}(x_i)}{1 - S_{ii}(X,k+1)} \right)^2 . \tag{17.8}$$

Ces deux écritures sont beaucoup plus intéressantes puisqu'elles ne nécessitent pas de calculer l'estimateur n fois, elles s'obtiennent directement à partir des matrices de lissage. Nous retrouvons le lien entre erreur d'estimation et erreur de prévision vu en régression où le terme $S_{ii}(X,\lambda)$ est remplacé par h_{ii}, le terme diagonal de la matrice de projection. En effet, dans l'exercice (3.4), nous montrons que l'erreur d'estimation $y_i - \hat{y}_i$ vaut l'erreur de prévision $y_i - \hat{y}_i^p$ divisée par $1 - h_{ii}$.

La fonction **npregress** propose différentes manières d'optimiser le paramètre par validation croisée, validation croisée généralisée ou leave-one-out. Pour choisir, il faut le préciser dans l'argument `cv.options` (argument de type liste).
Reprenons l'exemple de l'ozone

```
> hcv <- npregress(T12,O3)$bandwidth
> hcv
[1] 1.688373
```

En ce qui concerne la fonction **knn.reg**, elle permet de calculer cette erreur leave-one-out pour l'estimateur des plus proches voisins. Pour $k = 10$ avec les données de l'ozone, il suffit d'utiliser

```
> knn.reg(train=T12o,y=O3o,k=10)$PRESS/length(T12o)
[1] 287.6629
```

On peut maintenant utiliser ce procédé pour sélectionner une valeur de k. On choisit d'abord une grille de candidats

```
> K_cand <- 1:49
```

On calcule ensuite l'erreur leave-one-out pour chaque valeur de la grille

```
> loo <- rep(0,length(K_cand))
> for (i in 1:length(K_cand)){
+     loo[i] <- knn.reg(train=T12o,y=O3o,
                        k=K_cand[i])$PRESS/length(T12o)
+ }
```

On choisit enfin la valeur de k qui a la plus petite erreur

```
> K_cand[which.min(loo)]
[1] 8
```

On choisira donc ici 8 plus proches voisins, choix qui est confirmé par la validation 10 blocs (voir exercice 17.7).
On peut maintenant tracer les deux estimateurs avec les paramètres sélectionnés :

```
> mod.kppv <- knn.reg(train=T12o,test=as.matrix(x),y=O3o,k=8)
> mod.noyau <- npregress(T12o,O3o,bandwidth = hcv)
> prev.noyau <- predict(mod.noyau,newdata=x)
> plot(T12o,O3o,pch=20,xlab="T12",ylab="O3")
> lines(x,mod.kppv$pred,col="blue",lwd=2)
> lines(x,prev.noyau,col="red",lty=2,lwd=2)
```

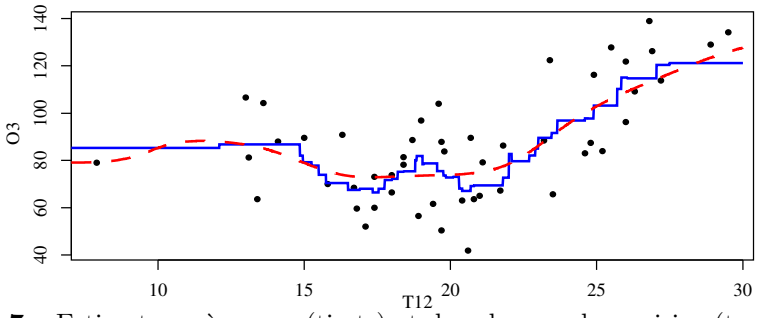


Fig. 17.7 – Estimateurs à noyau (tirets) et des plus proches voisins (trait plein) pour les valeurs de h et k sélectionnées par validation croisée.

Une fois le paramètre λ choisi, peut-on comparer cette estimation à celle obtenue avec la régression spline ou à la régression multiple ? Avec ces dernières, il est facile de contrôler le nombre de paramètres utilisés (5 dans l'exemple de la régression spline pour la concentration de l'ozone, voir en fin de section 16.2.2, p. 386). Avec les estimations non paramétriques, il faut passer par le nombre de degrés de liberté équivalent. Nous avons ici

```
> mod.noyau$df
5.34
```

Le nombre de degrés équivalent est 5.34. Nous avions estimé cette même fonction avec 5 paramètres avec des B-splines et la sortie du code R donnant les résultats des splines de lissage indiquait un nombre de degrés de liberté équivalent de 4.16. Pour les k plus proches voisins la trace de la matrice de lissage vaut n/k. Par conséquent, le nombre de degrés de liberté équivalent est de $50/8 = 6.25$.

17.4 Ecriture multivariée et fléau de la dimension

17.4.1 Ecriture multivariée

Dans cette section nous sommes en présence de p variables explicatives $X_1, X_2, \ldots, X_p$ et nous avons le modèle suivant

$$y_i = f(x_i) + \varepsilon,$$

où $x_i \in \mathbb{R}^p$ et où $\mathbb{E}(\varepsilon_i) = 0$.

– L'estimateur à noyau (de f) est défini de manière générale par

$$\hat{f}_h(x) = \frac{\sum_{i=1}^n y_i K_h\left(x - x_i\right)}{\sum_{i=1}^n K_h\left(\frac{x-x_i}{h}\right)},$$

où $h \in \mathbb{R}^{+p}$ est un vecteur de fenêtres, et la fonction K_h est définie par

$$\begin{aligned} \mathbb{R}^p &\to \mathbb{R}^+ \\ (z_1, \ldots, z_p) &\mapsto K_h(z_1, \ldots, z_p) = K^{(p)}\left(\tfrac{z_1}{h_1}, \ldots, \tfrac{z_p}{h_p}\right) \end{aligned}$$

où $K^{(p)}$ est un noyau, c'est-à-dire une fonction $K^{(p)} : \mathbb{R}^p \to \mathbb{R}^+$ dont l'aire sous la courbe vaut 1.
Cette écriture très générale peut être simplifiée. Tout d'abord nous allons ici considérer que les fenêtres sur chaque axe (noté h_j) sont constantes et égale à h. Remarquons qu'il est important de standardiser chaque variable lorsqu'on fait ce choix. Ensuite, plutôt que d'utiliser un noyau multivarié on se ramène à un noyau univarié, soit en proposant un produit de noyau, soit en utilisant une norme (en général euclidienne) :

$$K^{(p)}(x_1, \ldots, x_p) = K(\|x\|),$$

où $K : \mathbb{R}^+ \to \mathbb{R}^+$ est un noyau univarié (uniforme, Gaussien...). Ce choix suppose implicitement que la notion de proximité est de symétrie sphérique. Avec ces hypothèses simplificatrices nous obtenons l'estimateur à noyau :

$$\hat{f}_h(x) = \frac{\sum_{i=1}^n y_i K\left(\frac{\|x - x_i\|}{h}\right)}{\sum_{i=1}^n K\left(\frac{\|x - x_i\|}{h}\right)}. \tag{17.9}$$

– L'estimateur des k plus proches voisins est défini très simplement : étant donné $(x_1, y_1), \ldots, (x_n, y_n)$ avec $x_i \in \mathbb{R}^p, y_i \in \mathbb{R}$ et $x \in \mathbb{R}^p$:

1. on réordonne les observations en fonction de leur proximité avec x :

$$(x_{(1)}(x), y_{(1)}(x)), \ldots, (x_{(n)}(x), y_{(n)}(x))$$

avec $\|x_{(1)}(x) - x\| \leq \ldots \leq \|x_{(n)}(x) - x\|$.

2. l'estimateur des k plus proches voisins s'obtient en faisant la moyenne des y_i qui figurent parmi les k plus proches voisins de x :

$$\hat{f}_k(x) = \frac{1}{k} \sum_{i=1}^k y_{(i)}(x).$$

17.4.2 Biais et variance

Il existe de nombreux résultats théoriques sur les estimateurs présentés dans la section précédente. Nous nous contenterons d'en présenter certains sans énoncer les hypothèses techniques. On pourra consulter Tsybachov (2003) et Biau & Devroye (2015) pour plus de détails. On rappelle que nous sommes dans le modèle de régression

$$y_i = f(x_i) + \varepsilon_i, \quad i = 1, \ldots, n$$

où les ε_i sont i.i.d d'espérance nulle et de variance $\sigma^2 > 0$, les x_i sont dans $\mathbb{R}^p$ et $f : \mathbb{R}^p \to \mathbb{R}$ est la fonction à estimer. On considère ici l'estimateur à noyau $\hat{f}_h(x)$ en $x \in \mathbb{R}^p$ défini par (17.9). On rappelle que son erreur quadratique moyenne se décompose en un terme de biais et un terme de variance :

$$\begin{aligned}\mathrm{EQM}(\hat{f}_h(x)) =& \mathbb{E}((\hat{f}_h(x) - f(x))^2 \\ =& \Big(\mathbb{E}(\hat{f}_h(x)) - f(x)\Big)^2 + \mathrm{V}(\hat{f}_h(x)).\end{aligned}$$

On s'intéresse tout d'abord au terme de variance. Sous certaines hypothèses techniques, garantissant notamment que les observations sont bien réparties dans l'espace (voir exercice 17.4), on montre qu'il existe des constantes c_1 et C_1 telles que

$$\frac{c_1}{nh^p} \leq \mathrm{V}[\hat{f}_h(x)] \leq \frac{C_1}{nh^p}.$$

Cette inégalité met en évidence l'influence de la valeur de la fenêtre h sur la variance de l'estimateur : un choix trop faible de h conduira à un estimateur possédant une forte variance alors qu'une forte valeur de h donnera une variance faible.

L'étude du terme de biais nécessite des hypothèses de régularité sur la fonction f, ces hypothèses consistent généralement à supposer l'existence de dérivées à certains ordres. Supposons ici que f est lipschitzienne de constante L, c'est-à-dire que $\forall (x_1, x_2) \in \mathbb{R}^p \times \mathbb{R}^p$

$$|f(x_1) - f(x_2)| \leq L\|x_1 - x_2\|,$$

alors on montre (voir exercice 17.4) que le carré du biais de l'estimateur à noyau est contrôlé par $C_2 h^2$:

$$\Big(\mathbb{E}(\hat{f}_h(x)) - f(x)\Big)^2 \leq C_2 h^2.$$

Ici encore on remarque l'influence de h : une forte valeur de h aura tendance à donner un biais élevé et réciproquement. Cependant il n'y a pas de dépendance de la dimension p dans cette borne concernant le biais.

En mettant ensemble toutes ces briques, il est possible de borner l'EQM d'un estimateur à noyau

$$\mathbb{E}(\hat{f}_h(x) - f(x))^2 \leq \frac{C_1}{nh^p} + C_2 h^2.$$

Cette borne peut être minimisée par rapport à h et la valeur de h permettant le minimum est de l'ordre de

$$h = h_n \sim n^{-1/(2+p)}.$$

Avec ce choix, nous concluons que

$$\mathbb{E}(\hat{f}_h(x) - f(x))^2 \leq \tilde{C} n^{-2/(2+p)}.$$

On retrouve d'un point de vue théorique l'importance du choix de ce paramètre. On peut visualiser cet importance sur la figure 17.5 de l'exemple de l'ozone. Nous voyons que l'estimateur avec la plus forte valeur de h ($h = 20$) est très lisse, très plat. Cet estimateur possède une variance faible mais un biais très élevé. A l'inverse, l'estimateur avec une très petite fenêtre ($h = 0.05$) est très instable. Cet estimateur est fortement dépendant des données : il a tendance à passer par quasiment tous les points de l'échantillon et donc à sur-ajuster les données. D'après les résultats ci-dessus, nous remarquons que cet estimateur possède un biais faible mais une variance très élevée. Le sur-ajustement se traduit généralement par un problème de variance des estimateurs.

Remarque
Nous avons présenté uniquement des résultats pour l'estimateur à noyau. On retrouve exactement le même type de propriétés pour l'estimateur des plus proches voisins. C'est cette fois le paramètre k qui régulera le compromis biais/variance. En effet, pour une forte valeur de k, on va considérer un très grand nombre de plus proches voisins. L'estimateur sera ainsi très (trop) stable, ce qui se traduira par peu de variance mais beaucoup de biais. A l'inverse, une très faible valeur de k va produire un estimateur très dépendant des données, qui risque de sur-ajuster (beaucoup de variance et peu de biais), voir figure 17.6. Ces intuitions se retrouvent mathématiquement. En effet, sous des hypothèses similaires à celles présentées pour l'estimateur à noyau, on montre (voir par exemple Biau & Devroye (2015)) que le carré du biais est de l'ordre C_1'/k et la variance de l'ordre de $C_2'(k/n)^{2/p}$ où C_1' et C_2' sont des constantes qui dépendent du modèle.

17.4.3 Fléau de la dimension

Nous proposons ici une discussion sur la différence entre les approches paramétriques et non paramétriques dans le modèle de régression. Poser un modèle revient à supposer que f appartient à une classe de fonctions $\mathcal{F}$. Par exemple, le modèle linéaire étudié dans les premiers chapitres du livre fait l'hypothèse que f est linéaire en ses composantes :

$$f(x) = f(x_1, \ldots, x_p) = \beta_1 x_1 + \ldots + \beta_p x_p.$$

Sous cette hypothèse, estimer f revient à estimer le vecteur $\beta = (\beta_1, \ldots, \beta_p)$ qui est de dimension finie, on parle de modèle paramétrique. A l'inverse, une approche non paramétrique fera l'hypothèse que f appartient à une classe de fonctions de dimension infinie, par exemple l'ensemble des fonctions continues ou dérivables ou encore lipschitziennes comme nous l'avons fait dans la section précédente. Présenté ainsi, l'approche non paramétrique peut sembler plus attractive. En effet, les hypothèses faites sur la fonction f sont généralement moins contraignantes dans les modèles non paramétriques (une fonction linéaire est par exemple forcément continue). Il y a néanmoins un prix à payer à utiliser des modèles non paramétriques. Ce prix se situe le plus souvent au niveau de la précision d'estimation. En effet, un modèle non paramétrique étant plus « grand » qu'un modèle paramétrique, il

va généralement être plus difficile de se rapprocher de la cible inconnue avec une approche non paramétrique. Les différences de précisions d'estimation peuvent le plus souvent se quantifier. Restons dans l'exemple du modèle (paramétrique) de régression linéaire. Nous avons dans les premiers chapitres étudié les propriétés des estimateurs des MCO. Il a notamment été montré que ces estimateurs étaient sans biais et que la variance était de l'ordre de $1/n$ (voir section 5.8.3, p. 114). On peut ainsi en déduire que l'erreur quadratique

$$\mathbb{E}(\|\hat{\beta}_{\text{MCO}} - \beta\|^2)$$

tend vers 0 à la vitesse $1/n$ (voir aussi exercice 17.5). Cette vitesse est courante pour les estimateurs paramétriques, on peut par exemple obtenir la même vitesse pour les estimateurs du maximum de vraisemblance du modèle logistique (voir chapitre 11). Si maintenant on fait simplement l'hypothèse (non paramétrique) que f est lipschitzienne, alors nous avons vu dans la section précédente que le carré du biais de l'estimateur à noyau est de l'ordre de h^2 et sa variance de l'ordre de $1/nh^p$. Comme la valeur de h qui minimise l'erreur quadratique est de l'ordre de $n^{-1/(p+2)}$ (voir exercice 17.4) on a pour cette valeur optimale

$$\mathbb{E}((\hat{f}_{h_{\text{opt}}}(x) - f(x))^2) \leq \frac{C}{n^{2/(p+2)}}. \tag{17.10}$$

Remarquons que cette vitesse dépend du nombre de variables explicatives p : elle diminue lorsque p augmente. Cela signifie que les approches non paramétriques seront généralement peu performantes lorsque p est grand : c'est le fléau de la dimension. Intuitivement ce phénomène peut s'expliquer par le fait que les estimateurs non paramétriques (noyau ou plus proches voisins par exemple) sont définis à partir des observations qui sont proches de x. Pour être précis, il faut donc suffisamment de données dans un voisinage de x. Or, lorsque la dimension augmente, les voisinages ont tendance à se vider, il y aura donc moins d'observations (et donc d'information) à disposition pour bien estimer.

Notons également que quel que soit p fixé, la vitesse (17.10) est plus lente que la vitesse paramétrique en $1/n$. On peut interpréter ce résultat en disant que les estimateurs non paramétriques sont moins précis que les estimateurs paramétriques.

En pratique, il est toujours difficile de savoir quelle modélisation retenir pour répondre à un problème posé. Le modèle paramétrique linéaire présente en effet l'avantage d'être plus facilement interprétable : on a un paramètre pour chaque variable mais il repose sur l'hypothèse que la relation entre Y et les covariables $X_1, \ldots, X_p$ est linéaire. Si cette hypothèse n'est pas vérifiée par les données, ce modèle ne sera pas performant. Il peut être intéressant d'utiliser les méthodes proposées au chapitre 10 pour comparer ces différentes familles de modèles.

Il existe des approches intermédiaires à ces deux types de modèles, on peut par exemple citer le modèle additif

$$y_i = f_1(x_{i1}) + \ldots + f_p(x_{ip}) + \varepsilon_i$$

où les $f_j, j = 1, \ldots, p$ sont des fonctions (inconnues) de $\mathbb{R}$ dans $\mathbb{R}$. Cette approche conserve la structure additive du modèle linéaire mais supprime la linéarité. Il

existe des procédures non paramétriques pour estimer les f_j, on peut par exemple citer l'algorithme du backfitting. Ces fonctions dépendant d'une seule variable, les estimateurs obtenus ne souffrent généralement pas du fléau de la dimension. On pourra consulter Wood (2006) pour un descriptif précis de ces modèles.

17.5 Exercices

Exercice 17.1 (Questions de cours)

1) En régression non paramétrique, si la fenêtre utilisée est petite, l'estimateur obtenu, en général,
   A. varie beaucoup,
   B. ne varie pas,
   C. il n'y a pas de rapport entre la variation de l'estimateur et la taille de la fenêtre.
2) Lors de l'utilisation de k-ppv, lorsque k est grand, l'estimateur obtenu, en général,
   A. varie beaucoup,
   B. ne varie pas,
   C. il n'y a pas de rapport entre la variation de l'estimateur et le nombre de voisins.
3) Lorsque l'on augmente le paramètre de lissage (fenêtre ou nombre de voisins), le degré de liberté équivalent
   A. diminue,
   B. augmente,
   C. il n'y a pas de rapport entre le degré de liberté équivalent et le paramètre de lissage.

Exercice 17.2 (Estimateur de Nadaraya-Watson)

Nous souhaitons effectuer une régression constante locale, cela revient à minimiser

$$\min_{\beta_1} \sum_{i=1}^{n} (y_i - \beta_1)^2 p_i(x),$$

où

$$p_i(x) = K\left(\frac{x - x_i}{h}\right).$$

Montrer que l'estimateur de $\beta_1(x)$ est

$$\hat{\beta}_1(x) = \frac{\sum_{i=1}^{n} y_i p_i(x)}{\sum_{i=1}^{n} p_i(x)}.$$

Exercice 17.3 (†Polynômes locaux)

Il est souvent préférable d'effectuer une régression linéaire locale à la place d'une régression constante. Cela revient alors à minimiser

$$\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_1 - \beta_2(x_i - x))^2 p_i(x),$$

où

$$p_i(x) = K\left(\frac{x - x_i}{h}\right).$$

Montrer que l'estimateur de $\beta_1(x)$ est

$$\hat{\beta}_1(x) = \frac{\sum_{i=1}^n y_i q_i(x)}{\sum_{i=1}^n q_i(x)},$$

où

$$\begin{aligned} q_i(x) &= p_i(x)(S_2 - (x_i - x)S_1) \\ S_1 &= \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)(x_i - x) \\ S_2 &= \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)(x_i - x)^2. \end{aligned}$$

Utiliser les résultats du chapitre 4 et écrire $(X'\Omega^{-1}X)$ en fonction de S_1 et S_2.

Exercice 17.4 († Estimateur à noyau uniforme dans $\mathbb{R}^p$)
On considère le modèle de régression

$$y_i = m(x_i) + \varepsilon_i, \quad i = 1, \ldots, n$$

où $x_1, \ldots, x_n \in \mathbb{R}^p$ sont déterministes et $\varepsilon_1, \ldots, \varepsilon_n$ sont des variables i.i.d. d'espérance nulle et de variance $\sigma^2 < +\infty$. On désigne par $\|\,.\,\|$ la norme euclidienne dans $\mathbb{R}^p$. On définit l'estimateur localement constant de m en $x \in \mathbb{R}^p$ par :

$$\hat{m}(x) = \operatorname*{argmin}_{a \in \mathbb{R}} \sum_{i=1}^n (y_i - a)^2 K\left(\frac{\|x_i - x\|}{h}\right)$$

où $h > 0$ et pour $u \in \mathbb{R}, K(u) = \mathbf{1}_{[0,1]}(u)$. On suppose que $\sum_{i=1}^n K\left(\frac{\|x_i - x\|}{h}\right) > 0$.

1) Donner la forme explicite de $\hat{m}(x)$.

2) Montrer que

$$\mathrm{V}[\hat{m}(x)] = \frac{\sigma^2}{\sum_{i=1}^n K\left(\frac{\|x_i - x\|}{h}\right)}$$

et

$$\mathbb{E}[\hat{m}(x)] - m(x) = \frac{\sum_{i=1}^n (m(x_i) - m(x)) K\left(\frac{\|x_i - x\|}{h}\right)}{\sum_{i=1}^n K\left(\frac{\|x_i - x\|}{h}\right)}.$$

3) On suppose maintenant que m est lipschitzienne de constante L, c'est-à-dire que $\forall (x_1, x_2) \in \mathbb{R}^d \times \mathbb{R}^d$

$$|m(x_1) - m(x_2)| \leq L\|x_1 - x_2\|.$$

Montrer que

$$|\mathrm{biais}[\hat{m}(x)]| \leq Lh.$$

4) On suppose de plus qu'il existe une constante C_1 telle que

$$C_1 \leq \frac{\sum_{i=1}^n \mathbf{1}_{B_h}(x_i - x)}{n\mathrm{Vol}(B_h)},$$

où $B_h = \{u \in \mathbb{R}^p : \|u\| \leq h\}$ est la boule de rayon h dans $\mathbb{R}^p$ et $\mathrm{Vol}(A)$ désigne le volume d'un ensemble $A \subset \mathbb{R}^p$. Montrer que

$$\mathrm{V}[\hat{m}(x)] \leq \frac{C_2 \sigma^2}{nh^p},$$

où C_2 est une constante dépendant de C_1 et d à préciser.

5) Déduire des questions précédentes un majorant de l'erreur quadratique moyenne de $\hat{m}(x)$.
6) Calculer h_{opt} la valeur de h qui minimise ce majorant. Que vaut ce majorant lorsque $h = h_{opt}$? Comment varie cette vitesse lorsque p augmente ? Interpréter.

Exercice 17.5 (Vitesse de la régression univarié en design equi-espacé)
On considère le modèle de régression linéaire

$$Y_i = \beta x_i + \varepsilon_i, \quad i = 1, \ldots, n,$$

où $x_i = i/n$ et les ε_i sont i.i.d tels que $\mathbb{E}[\varepsilon_i] = 0$ et $\mathrm{V}(\varepsilon_i) = \sigma^2$.
1) Calculer l'estimateur des moindres carrés $\hat{\beta}$ de β.
2) Calculer le biais et la variance de $\hat{\beta}$.
3) En déduire que le risque quadratique de $\hat{\beta}$ vérifie

$$\mathbb{E}[(\hat{\beta} - \beta)^2] = \mathrm{O}\left(\frac{1}{n}\right).$$

Exercice 17.6 († Critère LOO)
On se place dans un modèle régression et on désigne par $\hat{f}_h$ et $\hat{f}_k$ les estimateurs à noyau de fenêtre $h > 0$ et des k plus proches voisins.
1) Montrer que ces deux estimateurs sont des lisseurs. On désignera par S_h et S_k les matrices de lissage.
2) Montrer que

$$\hat{f}_h^i(x_i) = \sum_{j \neq i} \frac{S_{ij,h}}{1 - S_{ii,h}} y_j$$

et

$$\hat{f}_k^i(x_i) = \sum_{j \neq i} \frac{S_{ij,k+1}}{1 - S_{ii,k+1}} y_j$$

où $\hat{f}_h^i$ et $\hat{f}_k^i$ sont les estimateurs à noyau et des k plus proches voisins calculés sans la i^{e} observation et $S_{ij,h}$ et $S_{ij,k}$ désignent les termes de la i^{e} ligne et la j^{e} colonne de S_h et S_k.
3) En déduire (17.7) et (17.8).

Exercice 17.7 (Caret et kppv)
Dans le cadre de l'explication de l'ozone (`O3`) par la température à midi (`T12`) nous utilisons un estimateur des k plus proches voisins.
Estimer k en utilisant une validation croisée 10 blocs en utilisant le package **caret**. On pourra utiliser la procédure ci-dessous.
1) Créer un data-frame `grille` contenant une variable nommée `k` contenant toutes les valeurs entières de 1 à 40.
2) Créer un objet (nommé `ctrl`) via la fonction **trainControl** pour proposer une validation croisée 10 blocs.
3) En utilisant `ctrl` et `grille` dans les arguments `trControl` et `tuneGrid` de la fonction `train` trouver le meilleur k au sens du RMSE.

Annexe A

Rappels

A.1 Rappels d'algèbre

Nous ne considérons ici que des matrices carrées réelles. Nous notons A une matrice et A' sa transposée. Pour i et j variant de 1 à n, nous noterons a_{ij} le terme courant de la matrice carrée A de taille $n \times n$ a_{ij}.

Quelques définitions

Une matrice A est *inversible* s'il existe une matrice B telle que $AB = BA = I$. On note $B = A^{-1}$.
La matrice carrée A est dite *symétrique* si $A' = A$,
singulière si $det(A) = 0$,
inversible si $det(A) \neq 0$,
idempotente si $AA = A$,
orthogonale si $A'A = AA' = I$.
définie positive si $x'Ax > 0$ pour tout $x \neq 0$.
semi-définie positive si $x'Ax \geq 0$ pour tout $x \neq 0$.

Le polynôme caractéristique est $\det(A - \lambda I)$. Les valeurs propres sont les solutions de $\det(A - \lambda I) = 0$. Un vecteur propre associé à la valeur propre λ est une solution non nulle de $Ax = \lambda x$.

Quelques propriétés

- $\mathrm{tr}(A) = \sum_{i=1}^{n} a_{ii}$.
- $\mathrm{tr}(A + B) = \mathrm{tr}(A) + \mathrm{tr}(B), \mathrm{tr}(AB) = \mathrm{tr}(BA)$ et $\mathrm{tr}(\alpha A) = \alpha\,\mathrm{tr}(A)$.
- $\mathrm{tr}(AA') = \mathrm{tr}(A'A) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}^2$.
- $\det(AB) = \det(A)\det(B)$.
- Si les matrices A et B sont inversibles, alors $AA^{-1} = A^{-1}A = I$, $(A^{-1})' = (A')^{-1}$,

$(AB)^{-1} = B^{-1}A^{-1}$ et $\det(A^{-1}) = 1/\det(A)$.
• La trace et le déterminant ne dépendent pas des bases choisies.

Matrices semi-définies positives (SDP)

• Les valeurs propres d'une matrice SDP sont toutes positives ou nulles (et réciproquement pour toute matrice symétrique).
• Si A est SDP et inversible, A est forcément définie positive (DP).
• Toute matrice A de la forme $A = B'B$ est SDP. En effet $\forall x \in \mathbb{R}^n$, $x'Ax = x'B'Bx = (Bx)'Bx = \|Bx\|^2 \geq 0$.
• Toute matrice de projecteur orthogonal est SDP. En effet, les valeurs propres d'un projecteur valent 0 ou 1.
• Si B est SDP, alors $A'BA$ est SDP.
• Si A est DP, B SDP alors $A^{-1} - (A+B)^{-1}$ est SDP.

Matrices symétriques

• Les valeurs propres de A sont réelles.
• Les vecteurs propres de A associés à des valeurs propres différentes sont orthogonaux.
• Si une valeur propre λ est de multiplicité k, il existe k vecteurs propres orthogonaux qui lui sont associés.
• La concaténation de l'ensemble des vecteurs propres orthonormés forme une matrice orthogonale P. Comme $P' = P^{-1}$, la diagonalisation de A s'écrit simplement $P'AP = \mathrm{diag}(\lambda_1, \cdots, \lambda_n)$.
• $\mathrm{tr}(A) = \sum_{i=1}^n \lambda_i$ et $\det(A) = \prod_{i=1}^n \lambda_i$.
• $\mathrm{rang}(A) =$ nombre de valeurs propres non nulles.
• Les valeurs propres de A^2 sont les carrés des valeurs propres de A et ces deux matrices ont les mêmes vecteurs propres.
• Les valeurs propres de A^{-1} (si cette matrice existe) sont les inverses des valeurs propres de A et ces deux matrices ont les mêmes vecteurs propres.

Propriétés sur les inverses

• Soit M une matrice symétrique inversible de taille $p \times p$ et u et v deux vecteurs de taille p. Nous supposerons que $u'M^{-1}v \neq -1$, alors nous avons l'inverse suivante

$$(M + uv')^{-1} = M^{-1} - \frac{M^{-1}uv'M^{-1}}{1 + u'M^{-1}v}. \tag{A.1}$$

• Soit M une matrice inversible telle que

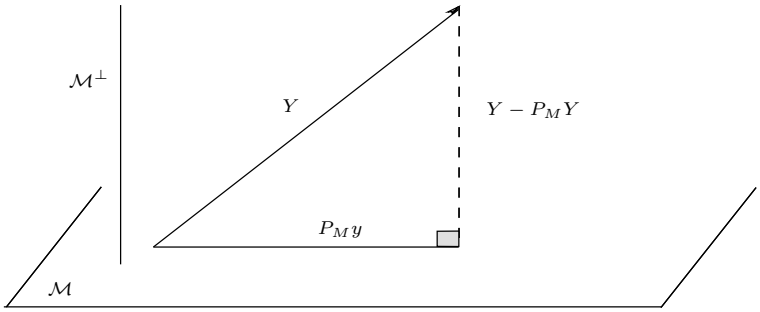
$$M = \left(\begin{array}{c|c} T & U \\ \hline V & W \end{array}\right)$$

avec T inversible, alors $Q = W - VT^{-1}U$ est inversible et l'inverse de M est

$$M^{-1} = \left(\begin{array}{c|c} T^{-1} + T^{-1}UQ^{-1}VT^{-1} & -T^{-1}UQ^{-1} \\ \hline -Q^{-1}VT^{-1} & Q^{-1} \end{array}\right).$$

Propriétés sur les projections

Une matrice carrée idempotente et symétrique est une matrice de projection orthogonale sur un sous-espace de $\mathbb{R}^n$, noté $\mathcal{M}$. P_M est un projecteur orthogonal, si le produit scalaire $\langle P_M y, y - P_M y\rangle = 0$ pour tout y de $\mathbb{R}^n$.
• Les valeurs propres d'une matrice idempotente valent 0 ou 1.
• Le rang d'une matrice idempotente est égal à sa trace.
• $\mathrm{tr}(P_M)$ est égal à la dimension de $\mathcal{M}$.
• La matrice $I - P_M$ est la matrice de projection orthogonale sur $\mathcal{M}^\perp$.



Soit $X = [X_1, \cdots, X_p]$ la matrice (n, p), de rang p, des p variables explicatives du modèle linéaire. Soit le sous-espace vectoriel $\Im(X)$ engendré par ces p vecteurs linéairement indépendants et P la matrice de projection orthogonale sur $\Im(X)$. Le vecteur $y - Py$ doit être orthogonal à tout vecteur de $\Im(X)$, or tous les vecteurs de $\Im(X)$ sont de la forme Xu, en particulier il existe un vecteur b tel que $Py = Xb$. Il faut donc que $\langle Xu, y - Py\rangle = 0$ pour tout vecteur u. En développant, nous obtenons $X'y = X'Py = X'Xb$. $X'X$ est inversible donc $b = (X'X)^{-1}X'y$ et donc $P = X(X'X)^{-1}X'$.

Dérivation matricielle

Soit f une fonction différentiable de $\mathbb{R}^p$ dans $\mathbb{R}$. Le gradient de f est par définition

$$\nabla(f) \quad = \quad grad(f) = \left[\frac{\partial f}{\partial u_1}, \cdots, \frac{\partial f}{\partial u_p}\right]$$

et le hessien de f est la matrice carrée de dimension $p \times p$, souvent notée $\nabla^2 f$ ou $H(f)$, de terme général $H(f)_{ij} = \frac{\partial^2 f}{\partial u_i \partial u_j}$.

Si $f(u) = a'u$ où a est un vecteur de taille p, alors $\nabla(f) = a'$ et $H(f) = 0$.
Si $f(u) = u'Au$, alors $\nabla(f) = u'(A + A')$ et $H(f) = A + A'$.

A.2 Rappels de probabilités

Généralités

Y vecteur aléatoire de $\mathbb{R}^n$ est par définition un vecteur de $\mathbb{R}^n$ dont les composantes $Y_1, \cdots, Y_n$ sont des variables aléatoires réelles. L'espérance du vecteur aléatoire Y, $\mathbb{E}(Y) = (\mathbb{E}(Y_1), \cdots, \mathbb{E}(Y_n))'$ est un vecteur de $\mathbb{R}^n$ et la matrice de variance-covariance de Y de taille $n \times n$ a pour terme général $\mathrm{Cov}(Y_i, Y_j)$.

$$\begin{aligned} \mathrm{V}(Y) = \Sigma_Y &= \mathbb{E}\left[(Y - \mathbb{E}(Y))\,(Y - \mathbb{E}(Y))'\right] \\ &= \mathbb{E}(YY') - \mathbb{E}(Y)(\mathbb{E}(Y))'. \end{aligned}$$

Considérons une matrice fixée (déterministe) A de taille $n \times n$ et b un vecteur fixé de $\mathbb{R}^n$. Soit Y un vecteur aléatoire de $\mathbb{R}^n$, nous avons les égalités suivantes

$$\begin{aligned} \mathbb{E}(AY + b) &= A\mathbb{E}(Y) + b \\ \mathrm{V}(AY + b) &= \mathrm{V}(AY) = A\,\mathrm{V}(Y)A'. \end{aligned}$$

Si Y est un vecteur aléatoire de $\mathbb{R}^n$ de matrice de variance-covariance Σ_Y, alors pour la norme euclidienne

$$\mathbb{E}(\|Y - \mathbb{E}(Y)\|^2) = \mathrm{tr}(\Sigma_Y).$$

Nous avons les égalités utiles suivantes

$$\mathrm{tr}(\mathbb{E}(YY')) = \mathbb{E}(\mathrm{tr}(YY')) = \mathbb{E}(\mathrm{tr}(Y'Y)) = \mathrm{tr}(\Sigma_Y) + \mathbb{E}(Y)'\mathbb{E}(Y).$$

Vecteurs aléatoires gaussiens

Un vecteur aléatoire Y est dit gaussien si toute combinaison linéaire de ses composantes est une v.a. gaussienne. Ce vecteur admet alors une espérance μ et une matrice de variance-covariance Σ_Y. On note $Y \sim \mathcal{N}(\mu, \Sigma_Y)$.
Un vecteur gaussien Y de $\mathbb{R}^n$ d'espérance μ et de matrice de variance-covariance Σ_Y inversible admet pour densité la fonction

$$f(Y) = (\frac{1}{2\pi})^{n/2} \frac{1}{\sqrt{|\det(\Sigma)|}} \exp\left[-\frac{1}{2}(Y - \mu)'\Sigma^{-1}(Y - \mu)\right].$$

Les composantes d'un vecteur gaussien $Y = (Y_1, \cdots, Y_n)'$ sont indépendantes si et seulement si Σ_Y est diagonale.

Soit $Y \sim \mathcal{N}(\mu, \Sigma_Y)$, alors $(Y - \mu)'\Sigma^{-1}(Y - \mu) \sim \chi^2_n$.

Théorème A.1 (Cochran)
Soit $Y \sim \mathcal{N}(\mu, \sigma^2 I)$, $\mathcal{M}$ un sous-espace de $\mathbb{R}^n$ de dimension p et P_M la matrice de projection orthogonale de $\mathbb{R}^n$ sur $\mathcal{M}$. Nous avons les propriétés suivantes :
(i) $P_M Y \sim \mathcal{N}(P_M \mu, \sigma^2 P_M)$ *;*
(ii) les vecteurs $P_M y$ et $y - P_M y$ sont indépendants ;
(iii) $\|P_M Y - P_M \mu\|^2/\sigma^2 \sim \chi^2_p$.

Bibliographie

Albert A. & Anderson D. (1984). On the existence of maximum likelihood estimates in logistic regression models. *Biometrika*, **71**, 1–10.

Antoniadis A., Berruyer J. & Carmona R. (1992). *Régression non linéaire et applications*. Economica.

Bertsekas D.P. (2016). *Nonlinear programming*. Athena scientific, 2 ed.

Biau G. & Devroye L. (2015). *Lectures on the nearest neighbor method*. Springer.

Birkes D. & Dodge Y. (1993). *Alternative methods of regression*. J. Wiley & Sons.

Chawla V., Bowyer K., Hall L. & Kegelmeyer W. (2002). Smote : Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, **16**, 321–357.

Cleveland W.S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the american statistical association*, **74**, 829–836.

Clémençon S. & Vayatis N. (2009). On partitioning rules for bipartite ranking. Dans *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, réd. D. van Dyk & M. Welling, vol. 5 de *Proceedings of Machine Learning Research*, pp. 97–104. PMLR, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA. http://proceedings.mlr.press/v5/clemencon09a.html.

Cohen J. (1960). A coefficient of agreement for nominal scales. *Educ. Psycol. Meas.*, **20**, 27–46.

Collet D. (2003). *Modelling Binary Data*. Chapman & Hall/CRC.

Cook R.D. (1977). Detection of influential observation in linear regression. *Technometrics*, **19**, 15–18.

De Jong S. (1995). Pls shrinks. *Journal of chemometrics*, **9**, 323–326.

Dodge Y. & Rousson V. (2004). *Analyse de régression appliquée*. Dunod.

Droesbeke J.J., Fine J. & Saporta G. (1997). *Plans d'expériences : applications à l'entreprise.* Technip.

Efron B., Hastie T., Johnstone I. & Tibshirani R. (2004). Least angle regression. *The annals of statistics*, **32**, 407–499.

Efron B. & Morris C.N. (1973). Stein's estimation rule and its competitors – an empirical bayes approach. *Journal of the american statistical association*, **68**, 117–130.

Efron B. & Tibshirani R. (1993). *An introduction to the bootstrap.* Chapman & Hall/CRC.

Eubank R. (1999). *Nonparametric regression and spline smoothing.* Dekker, 2 ed.

Fahrmeir L. & Kaufman H. (1985). Consistency and asymptotic normality of the maximum likelihood estimator in generalized linear models. *The annals of statistics*, **13**, 342–368.

Fu W.J. (1998). Penalized regressions : The bridge versus the lasso. *Journal of computational and graphical statistics*, **7**, 397–416.

Giraud C. (2014). *Introduction to High-Dimensional Statistics.* Chapman & Hall/CRC.

Golub G. & Van Loan C. (1996). *Matrix computations.* J. Hopkins Univ. Press.

Hastie T., Tibshirani R. & Friedman J. (2001). *The elements of statistical learning - data mining, inference and prediction.* Springer.

Hoaglin D. & Welsch R. (1978). The hat matrix in regression and anova. *The american statistician*, **32**, 17–22.

Hoerl A.E. & Kennard R.W. (1970). Ridge regression : biased estimation for nonorthogonal problems. *Technometrics*, **12**, 55–67.

Hosmer D.W. & Lemeshow S. (2000). *Applied Logistic Regression.* J. Wiley & Sons, 2 ed.

Huber P. (1981). *Robust Statistics.* J. Wiley & Sons.

Lehmann E.L. & Casella G. (1998). *Theory of point estimation.* Springer.

Lejeune M. (2004). *Statistique. La théorie et ses applications.* Springer.

Mallows C.L. (1973). Some comments on C_p. *Technometrics*, **15**, 661–675.

Mallows C.L. (1986). Augmented partial residuals. *Technometrics*, **28**, 313–319.

Miller A. (2002). *Subset selection in regression.* Chapmann & Hall/CRC, 2 ed.

Montgomery D.C., Peck E.A. & Vining G.G. (2001). *Introduction to linear regression analysis.* J. Wiley & Sons, 3 ed.

Noël Y. (2015). *Psychologie statistique avec R.* EDP sciences.

Rousseeuw P.J. & Leroy A.M. (1987). *Robust regression and outlier detection.* J. Wiley & Sons.

Scheffé H. (1959). *The analysis of variance.* J. Wiley & Sons.

Schwarz G. (1978). Estimating the dimension of a model. *The annals of statistics*, **6**, 461–464.

Tibshirani R. (1996). Regression shrinkage and selection via the lasso. *Journal of the royal statistical society : series B*, **58**, 267–288.

Tomek I. (1976). Two modifications of cnn. *IEEE Transactions on Systems, Man, and Cybernetics*, **6**, 769–772.

Tsybachov A. (2003). *Introduction à l'estimation non-paramétrique.* Springer.

Upton G.J.G. & Fingleton B. (1985). *Spatial analysis by example*, vol. 1. J. Wiley & Sons, 2 ed.

Velleman P.F. & Welsh R.E. (1981). Efficient computing of regression diagnostics. *The american statistician*, **35**, 234–242.

Wood S.N. (2006). *Generalized Additive Models : An Introduction with R.* Chapmann & Hall/CRC.

Zou H. & Hastie T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society : series B*, **67**, 301–320.

Index

R_a^2, *voir* R^2 ajusté

A

Aberrant, 54
Accuracy, *voir* Critères classification binaire, accuracy
Affine, 9
AIC, 175, 281
Aire sous la courbe ROC, 314, 330
Aitken, *voir* Estimateur, d'Aitken
Ajustement, 63
  individuel, 53
Aléatoire
  bruit, 11, 32
  estimateur, 13
Alternée, *voir* Choix de variables, pas à pas (*stepwise*)
Analyse de la covariance, 119
Analyse de la variance, 131, 142
  à 1 facteur, 131
  à 2 facteurs, 142
  tableau, 137, 150
ANOVA, *voir* Analyse de la variance
Apprentissage-validation, 238
AUC, *voir* Aire sous la courbe ROC
Autocorrélation des résidus
  choix des résidus, 52
  régression avec, 84, 87
  vérification graphique, 57
  vérification par test, 57
Autorégressif
  d'ordre 1, AR(1), 84
  simultané SAR, 87

B

Backward, *voir* Choix de variables, descendant (*backward*)
Biais
  de sélection, 164
  d'un estimateur, 14
  équilibre biais-variance, 39, 40, 408
  estimateur à noyau, 407
  estimateur des k plus proches voisins, 409
  estimateur des MC, 14, 38
  estimateur du MV, 92
  estimateur ridge, 77
BIC, 175, 281
Bootstrap, 109
Bruit
  définition, *voir* Aléatoire, bruit
  estimation, *voir* Résidus, définition

C

Cas/témoins, 358
Centrage-réduction
  définition, 195
  utilité, 194
Centre de gravité du nuage, 12
Chemin de régularisation, 312, 318
Choix
  fenêtre du noyau, *voir* Fenêtre, choix
  nombre de voisins k, *voir* Nombre de voisins k, choix
Choix de variables, 159
  AIC, 175, 281
  algorithmes, 178, 280
  apprentissage-validation, 167
  ascendant (*forward*), 178, 282

BIC, 175, 281
C_p de Mallows, 172
de la logistique, 279
descendant (*backward*), 179, 282
du modèle de Poisson, 302
exhaustif, 178, 280
généralités, 168
package **bestglm**, 281, 302, 335, 342
package **leaps**, 223
pas à pas (*stepwise*), 179, 282
R^2, 170
R^2 ajusté, 171
test, 169
validation croisée, 168
vraisemblance, 174
Classe de fonctions, 9, 32
en escaliers, 396
linéaires, 10
lipschitzienne, 408
Codage, 118, 183, 200, 208, 296, 311
Coefficient de détermination, *voir* R^2
ajusté, *voir* R^2 ajusté
Colinéarité des variables explicatives, 74
Comparaison de modèles
Apprentissage-validation, 238
apprentissage-validation
définition, 238
fonction de perte, 237
validation croisée, 321
applications, 241–248, 334–340
choix de composantes, 221, 228
choix de variables, 167
définition, 239
GLM régularisé, 315
leave-one-out, 168, 240, 325, 404
noyau, 403
Composantes
PLS, 226
principales, 218
Confiance
ellipse, *voir* Ellipse de confiance
hyperbole, 21, 22, 25
intervalle, *voir* Intervalle de confiance
intervalle par bootstrap, *voir* Intervalle de confiance
région, *voir* Région de confiance
Contraintes
identifiantes, 133, 134, 145, 256, 296
norme, 205
norme l^2, 75
norme minimum, 155
sur les coordonnées, 50
Contrastes, 155
Convergence, 114
en loi, 115
estimateur, 114
Cook, 60
Courbe
lift, 331
ROC, 329, 338
Coût, *voir* Fonction, de coût
Covariance des erreurs
exemples, 84
généralités, 84
vérification graphique, 57
C_p de Mallows, 172
Critères classification binaire
accuracy, 326, 367
balanced accuracy, 367
F_1-score, 368
faux négatifs, 327
faux positifs, 327
kappa de Cohen, 368
recall, 327
sensibilité, 327
Spécificité, 327
Critères d'information
AIC, 175
BIC, 175
équivalences, 176
généraux, 176

D

Décentrée (loi), 101, 108, 113
Décomposition en valeurs singulières, 211
Degré de liberté, 43, 100
équivalent, 393, 404, 406
Dépendantes (variables), *voir* Variables, explicatives
Déviance
binomiale, 314
Poisson, 314
DFFITS, 63

Distance
de Cook, 60, 61
Welsh-Kuh, 63
Données déséquilibrées, 353

E
Ecart-type, *voir* Variance, résiduelle, définition
Echantillon
apprentissage-validation, *voir* Comparaison de modèles, Apprentissage-validation
observations, 11, 13
Echelle de mesure, 194
Elastic-net, 309–313
elastic-net, 310
Ellipse de confiance, 21, 26, 95
GNU-R, 95
Emboîtés (modèles), 98, 169, 176, 264
EQM
définition, 39
estimateur à rétrécissement, 212
estimateur ridge, 77, 86
modèle de régression, 164–166, 172
Valeurs ajustées, 164
EQMP, 166, 168, 221, 228
Equations normales, 11
Erreur
définition, *voir* Aléatoire, bruit
estimation, *voir* Résidus, définition
Erreur de classification, 314
Erreur de prévision, 16, 41
Erreur quadratique moyenne de prévision, 166
Espace
des observations, 16
des résidus, 35, 40
des solutions, 35
des variables, 17, 35
Estimable, *voir* Unicité, $\hat{\beta}$
Estimateur
à noyau, 399, 406
$\hat{\beta}$, *voir* Estimateur, MC
d'Aitken, 82
de James-Stein, 78, 203, 204
de variance minimale, 14, 15, 38
des k plus proches voisins, 402, 407
des moindres carrés, *voir* Estimateur, MC
des moindres carrés contraints, 113
MC, 34, 135
loi, 92, 93
variance, 14, 41
MCG, 82
MV, 91
de la logistique, 259
Poisson, 296
polynômes locaux, 411
sans biais, 14, 15, 41
σ^2
loi, 92, 93
Eucalyptus
ANCOVA, 117, 119, 129
introduction, 5
régression multiple, 45
régression simple, 26
tests, 104
Exogènes (variables), *voir* Variables, explicatives

F
F_1-score, *voir* Critères classification binaire, F_1-score
Faux négatifs, *voir* Critères classification binaire, faux négatifs
Faux positifs, *voir* Critères classification binaire, faux positifs
Features ingeniering, 245, 342
Fenêtre
choix, 403
définition, 399
Fictives, *voir* Variables, fictives (dummy)
Fonction
de coût, 5–7, 32
absolu, 8
absolu ou MAE, 7
quadratique, 7, 8
de lien
canonique, 291
classiques, 290
définition, 290
en R, 291

logistique, 254, 291
Poisson, 291
de perte, 5
fixe d'une variable
cas général, 34
exemple de l'eucalyptus, 45
Forward, *voir* Choix de variables, ascendant (*forward*)

G
Gauss-Markov, 15, 38, 80
Géométrie
espace des variables, 35
modèle de régression, 33
modèle de régression avec interaction, 33
modèle de régression avec interaction et carré, 34
régression, 35
régression ridge, 76
GLM, *voir* Modélisation, GLM
Group lasso, 317

H
Hétéroscédasticité, 55, 78, 79, 293
Homoscédasticité, 14, 38, 55, 78
Hypothèses
gaussiennes, 19, 91
$\mathcal{H}_1$
définition, 12, 33
$\mathcal{H}_2$
définition, 14, 38
non vérifiée, 73–88
vérification, 55–57
$\mathcal{H}_3$
définition, 19, 91
non vérifiée, 109–111
vérification, 54–57
maximum de vraisemblance (MV), 19, 91
moindres carrés (MC), 91

I
Identifiabilité, 133
contraintes, *voir* Contraintes, identifiantes
introduction, 133
Indice de Younden, 332
Individus, 16
aberrants, 54
influents, 9, 54
nombre, 33, 74, 114
Influent, 54
Interaction
ANCOVA, 119
ANOVA, 142
généralités, 33, 34
variable qualitative et constante, 122
variable quantitative et qualitative, 121
variables qualitatives, 144
Intervalle de confiance, 95
β, 20, 94
bootstrap, 109, 111
comparaison, 111
de la logistique, 261
de la régression de Poisson, 301
droite de régression, 21
GNU-R, 25, 95
prévision, 21, 97
σ^2, 20, 94
Inverse
estimateur des MC, 35
généralisé de Moore-Penrose, 155
problème d', 74, 133
IRLS, *voir* Score, de Fisher

J
James-Stein, *voir* Estimateur, de James-Stein

L
Lars, 213–216
Lasso, 195–210, 309–313
group, 317
Levier, 59
Linéaire, *voir* Classe de fonctions, linéaires
Lisseur, 55
Logistique, *voir* Modèle de régression, logistique

M
MAE, 221
Matrice
de lissage
k plus proches voisins, 403

noyau, 403
de projection, 36, 59
du plan d'expérience, 32
particulière, 415
Matrice de confusion, *voir* Table de confusion
Matrice de lissage, 393
Modèle de régression
de Poisson, 289
emboîtés, *voir* Emboîtés (modèles)
linéaire multiple, 32
linéaire simple, 10
logistique, 251
sur variables centrées-réduites, 194, 217, 225
Modèle saturé
pour la logistique, 270
Modélisation
GLM, 289
logistique, 252
Poisson, 294
Moindres carrés
généralisés, 79
ordinaires (MC), *voir* Estimateur, MC
pondérés, 79
Moore-Penrose, *voir* Inverse, généralisé de Moore-Penrose

N
Nadaraya-Watson, *voir* Estimateur, à noyau
Nombre de voisins k
choix, 403
définition, 402
Non linéaire
estimateur à noyau, *voir* Estimateur, à noyau
fonction fixe, 34, 45
recherche de fonction fixe, 66, 70
régression spline, *voir* Spline, régression
Normales, *voir* Equations normales
Normalité
asymptotique, 115
Noyau
application linéaire
théorème du rang, 155
unicité MC, 155
estimateurs à, *voir* Estimateur, à noyau

O
Observations, *voir* Individus
Orthogonales
variables explicatives, 37, 50, 186, 187, 203
Oversampling, *voir* Sur-échantillonnage
Ozone
ANCOVA, 124
ANOVA
1 facteur, 137
2 facteurs, 142, 150
bootstrap, 110
choix de variables, 180
introduction, 3
régression multiple, 43
régression simple, 22
tests, 102
validation du modèle, 67

P
Paramètre de régularisation, 313
Paramètres, 10
IC, *voir* Intervalle de confiance
nombre, 10, 32
nombre effectif, 404
noyau, 393, 406
spline, 393
PCR, 218–225
Pénalisation, 174
Plan d'expérience
complet, 143
équilibré, 143, 145
incomplet, 143
matrice du plan, 32
PLS
PLS1, 225–231
PLS2, 227
Poids, *voir* Régression, pondérée
Point levier, 59
Poisson, *voir* Modèle de régression, de Poisson
Polynômes
locaux, *voir* Estimateur, polynômes locaux
régression, *voir* Régression, polynomiale

Population, 54, 60, 121
Prévision
erreur, 16, 41, *voir* EQMP
intervalle, 21, 97
PCR, 223
PLS, 229
ponctuelle, 15, 41, 97
pour la logistique, 267
ridge
lasso, 195
variance de l'erreur, 16
Profil, 147
Projection orthogonale, 18, 36, 99, 417

Q
Q-Q plot, 55, 68
Qualité
ajustement
graphique, 24, 28, 44, 46
individuel, 53
numérique, 5, 7, 18, 44, 171
par variable, 63
estimateur, 19, *voir aussi* EQM, définition
modèle, 221, 228
prévision, 24, 28, 160

R
R^2, 18, 42, 170
R^2 ajusté, 43, 171
Random oversampling, 362
Recall, *voir* Critères classification binaire, recall
Rééquilibrage, 361
Région de confiance, 20, 94, 95
multivariée (ellipsoïde), 20
univariée (intervalle), 20
Règles de prévision, 321
Régression
biaisée, 74
lars, 213–216
lasso, 195–210
logistique, 251
Log-linéaire, 295
modèle, 10
multiple, 32
PLS, 225–231
Poisson, 289
polynomiale, 34, 381–385
pondérée, 81
ridge, 73–78, 195–210
robuste, 9, 58, 60–63
simple, 10
spline, 386–390
sur composantes principales, 218–225
Régularisation, 309
Résidus, 80
choix, 53
de la logistique, 275
déviance, 276
Pearson, 275
définition, 15, 40
normalisés, 52
partiels, 65
partiels augmentés, 65
représentations graphiques, 53–57, 65, 66
exemples, 67–70
standardisés, 52
standardisés par validation croisée (VC), 52
studentisés, *voir* Résidus, standardisés par validation croisée (VC)
théoriques, 52
Rétrécissement (*shrinkage*)
estimateur, 211, 212
James Stein, 204
ridge, 78, 203
Ridge, 73–78, 195–210, 309–313
ROC, *voir* Courbe, ROC

S
SAR, *voir* Autorégressif, simultané SAR
Score
de Fischer, 297
de Fisher, 259
fonction de, 258
Scoring, 314
définition, 322
performance, 329
Sélection de modèles, *voir* Choix de variables
Sensibilité, 327, *voir* Critères classification binaire, sensibilité

Shrinkage, *voir* Rétrécissement (*shrinkage*), estimateur
SMOTE, 362
Somme des carrés
  expliquée, 18, 42
  résiduelle, 18, 40, 42
  totale, 18, 42
Somme des valeurs absolues résiduelles, *voir* Fonction, de coût, absolu ou MAE
Sous-échantillonnage, 364
Spécificité, 327, *voir* Critères classification binaire, spécificité
Spline
  B-splines, 247, 388
  de lissage, 390–393
  nœuds, 387, 388
  puissances tronquées, 387
  régression, 247, 386–390
Strate, 142
Student
  équivalence avec test F, 101, 108
  test nullité β_j, 101
Sur-échantillonnage, 362

T
Table de confusion, 323–326, 366
  ajustée, 324
  apprentissage/validation, 324
  empirique, 323
  théorique, 323
  validation croisée, 324
Tableau d'analyse de la variance, 137, 150
Test, 98, 123
  de Wald, 265
  Déviance, 272
  entre modèles emboîtés, 98, 100, 169, 176, 264
  $F = T^2$, 101, 108
  Fisher global, 102
  Hosmer et Lemeshow, 274
  hypothèse linéaire, 100
  hypothèse linéaire quelconque, 108, 112
  Pearson, 272
  rapport de vraisemblance, 108, 265, 300
  robustesse, 137
Théorème du rang, 155
Tomek
  algorithme, 364
  lien, 364
Transformation d'une variable, *voir* Fonction, fixe d'une variable, cas général

U
Undersampling, *voir* Sous-échantillonnage
Unicité
  $\hat{\beta}$, 133, 140
  contraste, 156

V
Valeurs
  ajustées, 12
    définition, 35
    EQM, *voir* EQM, Valeurs ajustées
    variabilité, 21, 40, 97, 105, 160, 163
  prévues
    définition, 12, 41
    EQMP, *voir* EQMP
    variabilité, 21, 41, 105, 160
Validation croisée, 167, 168, 221, 228, *voir* Comparaison de modèles, validation croisée
  données déséquilibrées, 371
Variables
  à expliquer, 3
  aléatoires, 11, 32
  choix de, *voir* Choix de variables
  exogènes, *voir* Variables, explicatives
  explicatives, 3, 11, 32
  fictives (dummy), 118, 200, 311
  nombre de, 33, 74, 160
Variables dépendantes, *voir* Variables, explicatives
Variance
  analyse, *voir* Analyse de la variance
  $\hat{\beta}$, 14, 41
  décomposition, 142
  estimateur à noyau, 407

estimateur des k plus proches voisins, 409
inter, 142
intra, 142
résiduelle
définition, 15
estimateur, 40
estimation, 15, 52, 63
IC, 94, 96

Vraisemblance, 309
de la logistique, 257
estimateur, 19
estimateur du max. de, 91
hypothèses, 19
pénalisation, 174
Poisson, 296

W
Welsh-Kuh, 63

Notations

β Vecteur de $\mathbb{R}^p$ de coordonnées $(\beta_1, \ldots, \beta_p)$, page 32

$\hat{\beta}_{(i)}$ Estimateur de β dans le modèle linéaire privé de l'observation i, page 52

$\beta_{\bar{j}}$ Vecteur β privé de sa j^{e} coordonnée, page 64

$\mathrm{Cov}(X, Y)$ Covariance entre X et Y, *i.e.* $\mathbb{E}\{(X - \mathbb{E}(X))(Y - \mathbb{E}(Y))'\}$, page 14

$c_{n-p}(1-\alpha)$ Fractile de niveau $(1-\alpha)$ d'une loi de χ^2 à $(n-p)$ ddl, page 21

ddl Degré de liberté, page 20

$\mathbb{E}(X)$ Espérance de X, page 14

$\mathcal{F}_{p,n-p}$ Loi de Fisher à p ddl au numérateur et $(n-p)$ degrés de liberté au dénominateur, page 20

$f_{(p,n-p)}(1-\alpha)$ Fractile de niveau $(1-\alpha)$ d'une loi de Fisher à $(p, n-p)$ ddl, page 20

$\mathcal{H}_2$ $\mathbb{E}(\varepsilon_i) = 0$ pour $i = 1, \cdots, n$ et $\mathrm{Cov}(\varepsilon_i, \varepsilon_j) = \delta_{ij}\sigma^2$, page 38

I_n ou I Matrice identité d'ordre n ou d'ordre dicté par le contexte, page 38

i.i.d. Indépendants et identiquement distribués, page 91

$\Im(X)$ Image de X (matrice $n \times p$) sous-espace de $\mathbb{R}^n$ engendré par les p colonnes de X : $\Im(X) = \{z \in \mathbb{R}^n : \exists \alpha \in \mathbb{R}^p, z = X\alpha\}$, page 35

$\mathcal{N}(0, \sigma^2)$ Loi normale d'espérance nulle et de variance σ^2, page 19

P_X Matrice de projection orthogonale sur $\Im(X)$, page 35

$\Pr(Y \leq y)$ Probabilité que Y soit inférieur ou égal à y, page 189

R^2 Coefficient de détermination, page 18

SCE Somme des carrés expliquée par le modèle, page 18

SCR Somme des carrés résiduelle, page 18

SCT Somme des carrés totale, page 18

$\hat{\sigma}_{(i)}$ Estimateur de σ dans le modèle linéaire privé de l'observation i, page 52

$\mathcal{T}_{n-p}$	Loi de Student à $(n-p)$ degrés de liberté, page 20
$t_{n-p}(1-\alpha/2)$	Fractile de niveau $(1-\alpha/2)$ d'une loi $\mathcal{T}_{n-p}$, page 20
VC	Validation croisée, page 52
x'_i	i^{e} ligne de X, page 32
X_j	j^{e} colonne de X, page 32
$X_{\bar{j}}$	Matrice X privée de sa j^{e} colonne, page 64
$\hat{y}_i$	Ajustement de l'individu i, page 15
$\hat{y}_i^p$	Prévision de l'individu i, page 16
$\hat{y}_\xi^p$	Prévision de l'individu $x^\star$ dans le modèle ayant ξ variables explicatives, page 166
$\hat{Y}_\xi^p$	Prévision des $n^\star$ individus de la matrice $X^\star$ dans le modèle à ξ variables, page 166
$\hat{y}(x_\xi)$	Ajustement de l'individu i dans le modèle ayant ξ variables explicatives, page 164
$\hat{Y}(X_\xi)$	Ajustement des n individus de la matrice X dans le modèle à ξ variables, page 164

Fonctions et packages R

B
barplot, 305
boxplot, 342, 350
bs, 247, 346, 389

C
confint, 25, 95, 263, 288, 302
cooks.distance, 68
cv.gglasso, 319
cv.glmnet, 206, 209, 243, 315, 336, 340, 348, 373
cvsegments, 224, 230

E
ellipse, 25

F
fitted, 127, 141

G
generalhoslem, 274
gglasso, 318
glmnet, 200, 209, 311, 318, 336, 338, 348

I
influence.measures, 68
interaction.plot, 147

L
leaps, 223
lm, 22, 27, 44, 46, 49
logitgof, 274

M
matlines, 24, 28, 69, 106
model.matrix, 184, 200, 209, 243, 311
MSEP, 224, 230, 244
multi_metric, 369, 376

O
offset, 360

P
pcr, 224, 244
plot
- 3D, 43, 49
- chemin
  - régularisation, 200, 312
- choix λ CV, 207
- défaut, 22, 24–26, 28, 45–46, 67, 96, 106, 125, 127, 141, 251, 272, 277, 293, 338, 382, 385, 389, 391–392, 400, 402, 405
- lm, 68
- QQ plot, 55, 68
- regsubsets, 180

plsr, 230, 244
predict, 24, 28, 46, 69, 106, 110, 184, 208–209, 242–245, 269, 276, 335, 391, 400

R
RandOverClassif, 362
RandUnderClassif, 364
resid, *voir* residuals
residuals, 69, 110, 141
roc_auc, 374
rstudent, 67, 127

S
SmoteClassif, 363
summary, 22, 27
summary.lm, *voir* summary

T
TomekClassif, 364

U
UBL, 361

X
xyplot, 127, 141

Y
yardstick, 369, 374